1966-05

Vibration analysis of piping systems.

Kim, Yong Su

Monterey, California. U.S. Naval Postgraduate School

VIBRATION ANALYSIS OF PIPING SYSTEMS

YONG SU KIM

VIBRATION ANALYSIS

OF PIPING SYSTEMS

by

Yong Su Kim
Lieutenant, Republic of Korea Navy
B.S., Republic of Korea Naval Academy, 1960

Submitted in partial fulfillment
for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

UNITED STATES NAVAL POSTGRADUATE SCHOOL
May 1966

ABSTRACT

A digital computer program, originally developed by George E. Fink, capable of determining the in-and/or out-of-plane vibration frequencies of a single plane piping system, using the basic transfer method, is modified and augmented by the writer. The program is modified to use the writer's method to reduce the amount of calculation and is augmented to use Marguerre and Uhrig's modifying frequency determinant method and Pestel and Mahrenholtz's remainder method for higher natural frequencies.

An analysis is made of difficulties encountered with the original transfer method and utilizing the modified methods.

The program accepts branched systems and non-stiff intermediate supports. A discussion and an explanation of how to use the program are included.

TABLE OF CONTENTS

Chapter		Page
I	INTRODUCTION	5
	1-1 General Remarks	5
	1-2 Scope of Work	5
	1-3 Assumptions and Limitations	6
	1-4 Notation	7
II	TRANSFER METHODS AND NUMERICAL DIFFICULTIES	9
	2-1 The State Vector and Transfer Method	9
	2-2 Failure of Transfer Method	13
	2-3 Reduction in Calculation in Matrix Multiplication	15
	2-4 Pestel and Mahrenholtz's Remainder Method	16
	2-5 Modified Delta Method	25
	2-6 Intermediate Condition-Branch System	27
	2-7 Computer	34
III	DISCUSSION	35
	3-1 Accuracy of Methods and Assurance of Solution	35
	3-2 Limitations and Some Suggestions	36
BIBLIOGRAPHY		37
APPENDIX		
A	DESCRIPTION OF PROGRAM	38
	A-1 General Remarks	38
	A-2 Program Structure	39
	A-3 Program Nomenclature	42
B	INSTRUCTIONS FOR USE OF PROGRAM	45
	B-1 General Remarks	45
	B-2 Coordinate System	47
	B-3 Data Required	47
	B-4 Component Identification	48
	B-5 Branch Joining Angle	50
	B-6 Boundary Conditions	50
	B-7 Control Cards	50
	B-8 Data Deck Assembly	51
	B-9 Flexibility of Program VIPIPE	52
	B-10 Format of Data	57

APPENDIX		Page
C	FLOW DIAGRAM	61
	C-1 General Remarks	61
	C-2 Index of Flow Diagram	61
	C-3 Flow Diagram	62
D	PROGRAM LISTING	80
	D-1 Index	80
	D-2 Listing	81
E	ACCURACY OF METHODS AND SAMPLE PROBLEMS	117
	E-1 General Remarks	117
	E-2 Method Accuracy Analyses	117
	E-3 Other Sample Problems	125

CHAPTER I

INTRODUCTION

1-1 General remarks.

Today, in increasing measure, natural frequencies of a system capable of linear vibration are being determined with the help of the method of transfer matrices. This method is well explained in the recent book by E. Pestel and Leckie [3].*

This method which has been known for more than ten years, proceeds with particular simplicity especially with the use of digital computer. However, it meets with important numerical difficulties.

Numerical difficulties were observed by George E. Fink in his U. S. Naval Postgraduate School Master's thesis "Vibration analysis of piping systems" [4]. He used the straight-forward method of transfer matrices for calculation of the natural frequencies of vibrating piping systems.

The purpose of this paper is to deal with numerical difficulties and with the remedies which have been developed and also to investigate reduction in calculation.

1-2 Scope of work.

Numerical difficulties are encountered at higher natural frequencies and for systems having very stiff exterior springs. In examining these difficulties the paper by Marguerre and Uhrig was encountered [2]. This tends to explain difficulties both with the original transfer method and with the Pestel and Mahrenholtz's modified methods [1]. Although, Marguerre and Uhrig suggest several different methods for dealing with

*Numbers in brackets refer to the bibliography, page 37.

the difficulties, those which appear to be certain to eliminate the difficulties are so far outside the spirit of the transfer method that the advantage of this method would be lost. Therefore, it was decided to proceed with the transfer method to get as much good from it as one could.

The writer set out to investigate the remedies for failure at high frequency. The digital computer program "VIPIPE" presented in the Appendices was used to determine the natural frequencies of in-and out-of-plane vibration of a planar piping system based on the transfer method. This program was originally developed by Fink [4], but has been modified and augmented by the writer.

1-3 Assumptions and limitations.

1. System is conservative.

2. Distributed mass or/and lumped mass model is used. For the lumped mass model used in curved section (or in straight section) sufficient subsectioning is carried out to give acceptable answer.

3. The transfer method surely fails if the system has rigid exterior elements. The hangers are introduced as a support by a point matrix with appreciable linear and/or torsional compliance.

4. Program VIPIPE is limited to a system of a maximum of fifty components, twenty branches, and twenty hangers. It can only determine in- and/or out-of-plane natural frequencies of planar system.

5. Boundary conditions should be such that half of the state quantities are vanishing.*

*Refer to Appendix B.

1-4 Notation.

$[\ \ ]$	matrix.
$\{\ \ \}$	column matrix.
X,Y,Z	cartesian right handed coordinate system.
u,v,w	displacement in the XYZ direction respectively.
δ,ψ,ϑ	rotation about XYZ axis respectively.
M	bending moment.
N	normal force.
T	torque.
V	shear force.
Z_i	state vector at location i.
$[Z_i]$	state matrix at location i.
$[U_{i+1}]$	transfer matrix from i to i + 1.
$\Delta(\omega)$	frequency determinant.
$\Delta_m(\omega)$	modified determinant.*
[N]	non-vanishing state matrix of Z_o.
u_{ij}	element of [U].
[S]	spring matrix.
[S]	purged frequency determinant in the P-M method.*
[I]	unit matrix.
[O]	null matrix.
d	displacement vector.
[d]	displacement matrix.
p	internal force vector.
[p]	internal force matrix.

*In the text, the following abbreviations are used. M-D method stands for Modified Delta method and P-M method for Pestel and Mahrenholtz's remainder method. These will be explained in what follows.

$[G]$	coordinate transfer matrix.
$[R1]$, $[R2]$, $[R3]$	square sub-matrices.
V_0	column vector of every elements value of 1. (cf. p. 16).
$R(\omega)$	remainder.
X	correction factor vector.
P_0 , P_1 , P_2	assumed state quantities at the starting boundary.
X_0 , X_1 , X_2 Y_0 , Y_1 , χ_1 , χ_2	correction factors in the P-M method.
D_1 , D_2	purging factors in the M-D method.
λ	eigenvalue.
x	eigenvector.

CHAPTER II

TRANSFER METHODS AND NUMERICAL DIFFICULTIES

2-1 The state vector and transfer matrix.

A. State vector.

The state of vibration at location i, specified by state quantities, is described by the state vector Z_i. This vector is a column matrix whose elements completely describe the instaneous displacements, both rectilinear and angular, of that point from its quiescent position, as well as the corresponding rectilinear and angular forces in the member at the same point and at the same instant, i.e., those forces, if a cutting plane were passed through the point at the instant of interest, would have to be applied to each cut face to prevent relative motion between them.

In the case of planar piping system, the state vector has six elements, three pertaining to displacement, and three pertaining to force.

The state vectors, for the in- and out-of-plane cases respectively, as used in VIPIPE, are:

$$Z_{ip} = \begin{Bmatrix} u & \text{displacement in X direction.} \\ v & \text{displacement in Y direction.} \\ \vartheta & \text{slope in X-Y plane with respect to X asix.} \\ M_z & \text{moment about Z axis.} \\ V_y & \text{shear in Y direction.} \\ N & \text{normal force in X direction.} \end{Bmatrix} \qquad (2\text{-}1\text{-}1)$$

$$
Z_{0P} = \left\{ \begin{array}{ll}
\delta & \text{twist about X axis.} \\
T & \text{torque about X axis.} \\
-w & \text{(negative of) deflection in Z direction.} \\
\vartheta & \text{slope in X-Y plane about X axis.} \\
M_y & \text{moment about Y axis.} \\
V_z & \text{shear in Z direction.}
\end{array} \right. \qquad (2\text{-}1\text{-}2)
$$

Thus, the state vector in this application is of order 6x1, while the system state matrix* is 6x3 and the transfer matrix of a section is 6x6; however, in developing theory we will use the more general case 2rx1, 2rxr, and 2rx2r respectively. A local XYZ right handed cartesian coordinate system and the elements of the state vector are arranged as in Fig. 2-1-1.

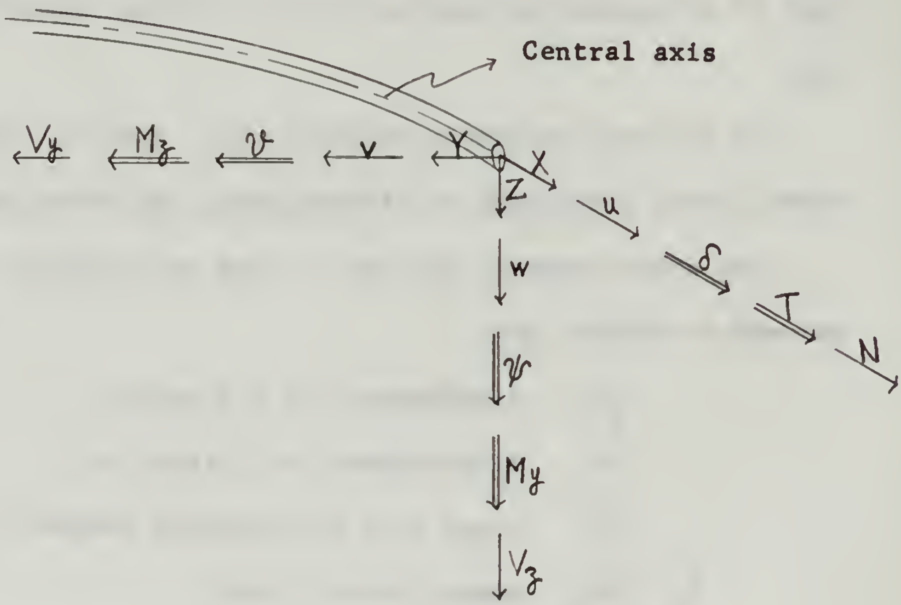


Fig. 2-1-1

B. Transfer matrix and transfer method.

From the state vector at location i (node i), the state

*The concept of "state matrix" is introduced in Section 2-3.

vector at location i + 1 is given by

$$Z_{i+1} = \left[U_{i+1}\right] \cdot Z_i \quad (i = 0,1,2,\ldots,n) \qquad (2\text{-}1\text{-}4)$$

Here, matrix $\left[U_{i+1}\right]$ relating the state vectors of node i and node i + 1 is termed a transfer matrix.

We number the node i + 1 immediately following the node i from the left end, the beginning of the main system. From the recurrence relation 2-1-4, one easily obtains

$$\begin{aligned} Z_n &= \left[U_n\right]\left[U_{n-1}\right] \cdot \cdot \cdot \left[U_3\right] \cdot \left[U_2\right] \cdot \left[U_1\right] \cdot Z_o \\ &= \left[U\right] \cdot Z_o \qquad (2\text{-}1\text{-}5) \end{aligned}$$

which is a linear relationship between the state quantities at location 0 and n of the system (boundaries of the main system). We call this method the transfer method.

Since of the 2r state quantities at each boundary r quantities are vanishing, only r are established by the boundary condition at location 0, and r homogeneous equations are established by the boundary condition at location n. The natural frequencies of the system are given by the zeros of the determinant of the corresponding r homogeneous equations. Since the application of interest is to the determination of the natural frequencies of piping systems, it would be instructive to illustrate the origin and nature of frequency dependence in transfer matrices. For this end, let us construct a point matrix for the in-plane case connecting Z_i^L with Z_i^R (Z_i^R and Z_i^L are state vectors of right side and left side, respectively, of a point mass m_i at node i).

To begin we assume that all of the elements of the state vectors vary with time, each having the factor cos ωt .

Noting that the deflection, slope, and moment* are continuous across the concentrated mass m_i, so that

$$u_i^R = u_i^L \quad , \quad v_i^R = v_i^L \quad , \quad \vartheta_i^R = \vartheta_i^L \quad , \quad M_i^R = M_i^L$$

From the free body diagram of Fig. 2-1-2A, B, the following relations are established.

$$u(t) = u.\cos\omega t \qquad \frac{d^2u(t)}{dt^2} = -u\omega^2\cos\omega t \qquad (2\text{-}1\text{-}6)$$

$$v(t) = v.\cos\omega t \qquad \frac{d^2v(t)}{dt^2} = -v\omega^2\cos\omega t \qquad (2\text{-}1\text{-}7)$$

$$V_i^R \cdot \cos\omega t - V_i^L \cos\omega t = -m\cdot\frac{d^2v(t)}{dt^2} \qquad (2\text{-}1\text{-}8)$$

$$N_i^R \cdot \cos\omega t - N_i^L \cdot \cos\omega t = -m\cdot\frac{d^2u(t)}{dt^2} \qquad (2\text{-}1\text{-}9)$$

and substituting Eqn. 2-1-6 into Eqn. 2-1-8 and Eqn. 2-1-7 into Eqn. 2-1-9 we have

$$V_i^R = V_i^L + mv\omega^2 \qquad (2\text{-}1\text{-}10)$$

$$N_i^R = N_i^L + mu\omega^2 \qquad (2\text{-}1\text{-}11)$$

Now, the relation of the state vectors of the right and left side of the point mass m in matrix form is:

$$Z_i^R = [U_P]\cdot Z_i^L$$

i.e.,

$$\begin{bmatrix} u \\ v \\ \vartheta \\ M_z \\ V_y \\ N \end{bmatrix}_{iR} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & m\omega^2 & 0 & 0 & 1 & 0 \\ m\omega^2 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} u \\ v \\ \vartheta \\ M_z \\ V_y \\ N \end{bmatrix}_{iL}$$

$[U_P]$ is a point matrix for in-plane vibration. It neglects any rotational inertia that the point mass may have; however, rotational inertia could

*The moment is continuous only if there are no point elements having finite rotational inertia.

be accounted for by including another nonzero term in the matrix.

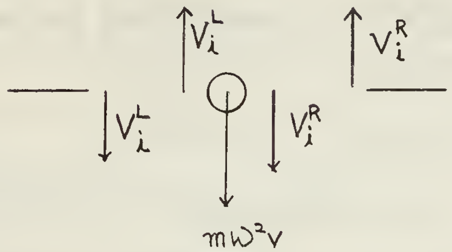


Fig. 2-1-2A

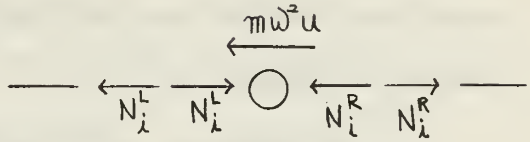


Fig. 2-1-2B

The derivation of transfer matrices is treated in great detail in reference [3] and transfer matrices for various components of planar piping system are incorporated in VIPIPE.

2-2 Failure of transfer method.

For higher natural frequencies and for systems having very stiff exterior springs numerical difficulties are encountered. In these cases, the numerical value of the frequency determinant is given as the difference between large numbers. So practically the natural frequencies can no longer be determined.

From a mathematical point of view, in case the transfer matrix $[U]$ has a single dominant real eigenvalue, the columns of $[U]^k$ become more and more parallel as k increases, and approach the first eigenvector x_1 [6],[2] . In this case the transfer matrix degenerates.

$$[U]^k x_1 = \lambda_1 x_1 \qquad (2\text{-}2\text{-}1)$$

This could be the case for the lumped mass system as the number of lumped masses increases. This was observed in the straight section lumped mass system, but for the curved lumped mass system this was not observed.

As the frequency increases (elements of some transfer matrices are functions of frequency), the same phenomenon was observed in the distributed mass straight section transfer matrix. For example, for the sample problem of Appendix E-2-1,

At frequency of 20 rad/sec,

the largest eigenvalue: $\lambda_1 = .11484022\text{x}10^{2}$

second largest eigenvalue: $\lambda_2, \lambda_3 = .99980414 \pm j0.01979160$

transfer matrix [U] is:

.99980413	0	0	0	0	$.90535567\backslash^{-5*}$
0	$.25105587\backslash^{1}$	$.25986435\backslash^{3}$	$.22549093\backslash^{-2}$	$.14255229\backslash^{0}$	0
0	$.30840073\backslash^{-1}$	$.25105587\backslash^{1}$	$.26651055\backslash^{-4}$	$.22549093\backslash^{-2}$	0
0	$.47565399\backslash^{4}$	$.30070198\backslash^{6}$	$.25105587\backslash^{1}$	$.25986435\backslash^{3}$	0
0	$.56219618\backslash^{2}$	$.47565399\backslash^{4}$	$.30840073\backslash^{-1}$	$.25105589\backslash^{1}$	0
$-.43265609\backslash^{2}$	0	0	0	0	.99980413

$*\backslash^{-5}$ means 10^{-5}, etc.

In this case, that is, at this frequency, we can see that there is no phenomenon of the columns becoming parallel. As the frequency increases this single dominant real eigenvalue keeps increasing, and at frequency of 1700 rad/sec.

the largest eigenvalue: $\lambda_1 = .5937220 \times 10^{10}$

second largest eigenvalue: $\lambda_2, \lambda_3 = -.11136847 \pm j.99377918$

transfer matrix [U] is:

-.11136847	0	0	0	0	.53482189\-5
0	.14843074\10	.13191200\11	.12023306\5	.10685242\6	0
0	.16701804\9	.14843074\10	.13528929\4	.12023306\5	0
0	.18324149\15	.16284870\16	.14843074\10	.13191200\11	0
0	.20618799\14	.18324149\15	.16701804\9	.14843074\10	0
-.18465906\6	0	0	0	0	.11136847

Normalized values of each column of $[U]$ are:

.6031032\-7	0	0	0	0	-.4802273\-5
0	.8100280\-6	.8100279\-6	.8100280\-6	.8100280\-6	0
0	.9114641\-7	.91146407\-7	.9114641\-7	.9114641\-7	0
0	1	1	1	1	0
0	.1125225\-1	.1125225\-1	.1125225\-1	.1125225\-1	0
1	0	0	0	0	1

We can see several of the columns are almost parallel to each other. From this, it is easy to see the numerical difficulties we should encounter. This depends on the nature of the transfer matrix. For systems made of various sections, multiplications of different transfer matrices may result in the difficulties taking care of themselves.

2-3 Reduction in calculation in matrix multiplication.

In equation 2-1-5, $[U] = [U_\eta] \cdot [U_{\eta-1}] \cdots [U_2] \cdot [U_1]$, so we should perform matrix multiplication of 2rx2r matrix times 2rx2r matrix successively.

Now using the fact of the boundary condition, introduce Z_0 as 2rxr matrix called the "state matrix." This is formed from the state

vector as follows, all elements of the state matrix are zero except that in its ith column, unity appears in the same position as the ith generally nonzero term reading down the state vector. Then we can multiply 2rx2r matrices by 2rxr matrices and thus can reduce the calculation considerably.

For example, in the in-plane case, for a **built-in** left end

$$Z_0 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The frequency determinant is obtained in the usual way from the boundary conditions at the right end.

We use the term "Delta method" to refer to this method which was discovered by the writer during the course of this study. This procedure is used in what follows instead of the usual procedure of the transfer method.

2-4 Pastel and Mahrenholtz's remainder method.

Instead of the frequency determinant, Pestel and Mahrenholtz introduced a remainder [1],[3] . This method, presenting the system state matrix as 2rxr matrix, reduces the amount of calculation considerably compared to the original transfer method, and claims greater accuracy at higher frequencies because the remainder, which is the difference between small numbers, is itself small. This will be explained later.

A. Remainder.

For the non-vanishing state quantities of the state vector Z_o

which depends on frequency ω we use estimated values which approximate the true values, following which the Z_i ($i = 1, 2, 3, \ldots, n$) are determined.

In the state vector Z_0, since for eigenvibration only the ratios of state quantities are of interest, one of the non-vanishing state quantities is assigned the value of 1.

For example, for the in-plane case, at certain frequency, a cantilevered pipe which is free at the left end and fixed at the right end has:

$$Z_0^0 = \begin{bmatrix} 1 \\ P_1^0 + X_1^0 \\ P_2^0 + X_2^0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad (2\text{-}4\text{-}1')$$

(in what follows, superscripts 0 and 1 denote the round of calculation; i.e., super-script 0 means first round of calculation with approximated state quantities in Z_0, and superscript 1 means second round of calculation with improved state quantities in Z_0). In equation 2-4-1', we assigned the value of 1 to the non-vanishing state quantity in the first row of Z_0, and the quantities of P_1^0, P_2^0 are the previously chosen approximations and X_1^0, X_2^0 are correction factors which are unknowns.

The state vector Z_0^0 of (2-4-1') in other matrix form is:

$$Z_0^0 = \begin{bmatrix} 1 & 0 & 0 \\ P_1^0 & 1 & 0 \\ P_2^0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ X_1^0 \\ X_2^0 \end{bmatrix} \qquad (2\text{-}4\text{-}1'')$$

For the actual matrix multiplication Z_0^o of (2-4-1'') is used. After a round of calculation, the approximated state quantities P_1^o , P_2^o will be improved in the form

$$
\begin{aligned}
P_1^1 &= P_1^o + \chi_1^o \\
P_2^1 &= P_2^o + \chi_2^o
\end{aligned}
\tag{2-4-2}
$$

where χ_1^o and χ_2^o are corrections obtained by this process; the difference between χ_1^o and X_1^o , χ_2^o and X_2^o will be shown later.

Then, the improved state vector Z_0^1 will be:

$$
Z_0^1 = \begin{bmatrix} 1 \\ P_1^1 \\ P_2^1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{2-4-3}
$$

When we start first, usually P_1^o , P_2^o are set equal to 1, in the absence of better information. From the same method as (2-1-5) using Z_0^o of (2-4-1'').

$$
Z_n^o = [U_n] \cdot [U_{n-1}] \ldots [U_2] \cdot [U_1] \cdot Z_0^o
$$

From the boundary condition at location n, i.e., from the three vanishing elements of Z_n , we get

$$
0 = \Delta(\omega) \cdot [N^o] \cdot X^o = [S] \cdot X^o \tag{2-4-4}
$$

$[N]$: nonvanishing sub-matrix of $[Z_0]$.

X : correction factor vector.

$\Delta(\omega)$: frequency determinant.

$[S]$: $\Delta(\omega) \cdot [N]$

For the cantilevered pipe of the example in the in-plane case, the frequency determinant $\Delta(\omega)$, see Ref. [3] , consists of the top left

3x3 submatrix of the transfer matrix [U] .

Equation 2-4-4 may be written:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ P_1^0 & 1 & 0 \\ P_2^0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ X_1^0 \\ X_2^0 \end{bmatrix}$$

$$= \begin{bmatrix} u_{11} + u_{12}P_1^0 + u_{13}P_2^0 & u_{12} & u_{13} \\ u_{21} + u_{22}P_1^0 + u_{23}P_2^0 & u_{22} & u_{23} \\ u_{31} + u_{32}P_1^0 + u_{33}P_2^0 & u_{32} & u_{33} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ X_1^0 \\ X_2^0 \end{bmatrix}$$

$$= \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ X_1^0 \\ X_2^0 \end{bmatrix} \tag{2-4-5}$$

In equation 2-4-5, the first column of [S] no longer contains large numbers, but the second and third columns remain unchanged. We call this "purging" the first column. From equation 2-4-5 we may write,

$$-\begin{bmatrix} s_{11} \\ s_{21} \\ s_{31} \end{bmatrix} = \begin{bmatrix} s_{12} & s_{13} \\ s_{22} & s_{23} \\ s_{32} & s_{33} \end{bmatrix} \cdot \begin{bmatrix} X_1^0 \\ X_2^0 \end{bmatrix} \tag{2-4-6}$$

Equation 2-4-6 has first, three equations with two unknowns (two correction factors), and second it has the purged column to the left and unpurged columns as coefficient matrix.

Equation 2-4-6 is the same as;

$$-\begin{bmatrix} u_{11} \\ u_{21} \\ u_{31} \end{bmatrix} = \begin{bmatrix} u_{12} & u_{13} \\ u_{22} & u_{23} \\ u_{32} & u_{33} \end{bmatrix} \cdot \begin{bmatrix} P_1^0 + X_1^0 \\ P_2^0 + X_2^0 \end{bmatrix} \tag{2-4-7}$$

From 2-4-7 we choose any two equations. If we choose the first two, then by Cramer's rule;

$$X_1^\circ = \frac{-\begin{vmatrix} u_{11} & u_{13} \\ u_{21} & u_{23} \end{vmatrix}}{\begin{vmatrix} u_{12} & u_{13} \\ u_{22} & u_{23} \end{vmatrix}} - P_1^\circ = \frac{\Delta_2}{\Delta_1} - P_1^\circ \qquad X_2^\circ = \frac{-\begin{vmatrix} u_{12} & u_{11} \\ u_{22} & u_{21} \end{vmatrix}}{\begin{vmatrix} u_{12} & u_{13} \\ u_{22} & u_{23} \end{vmatrix}} - P_2^\circ = \frac{\Delta_3}{\Delta_1} - P_2^\circ \tag{2-4-8}$$

The Δ 's are defined by the preceeding expressions. If $\omega = \omega_k$, where ω_k is a natural frequency of the system, then the third equation will be satisfied if we substitute these values for X_1° and X_2°. Otherwise, substituting the above value for X_1°, we get, from the third equation, a somewhat different value for X_2°; we denote this by the symbol Y_2°.

$$Y_2^\circ = -\frac{u_{32}\Delta_2}{u_{33}\Delta_1} - \frac{u_{31}}{u_{33}} - P_2^\circ \tag{2-4-9}$$

We define the remainder $R^\circ(\omega)$ to be the difference between these values.

$$R^\circ(\omega) = X_2^\circ - Y_2^\circ = \frac{\Delta_3}{\Delta_1} + \frac{u_{32}\Delta_2}{u_{33}\Delta_1} + \frac{u_{31}}{u_{33}} \tag{2-4-10}$$

As we see from equation 2-4-10, for fixed ω, the remainder is independent of the assumed state quantities P_1° and P_2°. If the remainder is zero, $\omega = \omega_k$ since all the prescribed boundary conditions will be satisfied simultaneously.

For the next round of calculation for the same frequency, form the arithmatic average of the two values X and Y for the correction factor χ, i.e.,

$$\chi_1^\circ = X_1^\circ \qquad \chi_2^\circ = \frac{1}{2}(X_2^\circ + Y_2^\circ)$$

Then, new starting values are:

$$P_1^1 = P_1^0 - X_1^0 = \frac{\Delta_2}{\Delta_1}$$

$$P_2^1 = P_2^0 - X_2^0 = \frac{1}{2}\left\{ \frac{\Delta_3}{\Delta_1} - \left(\frac{U_{32}\Delta_2}{U_{33}\Delta_1} + \frac{U_{31}}{U_{33}} \right) \right\} \qquad (2\text{-}4\text{-}11)$$

which are also independent of P_1^0 and P_2^0. A repeated round of calculation gives

$$X_1^1 = X_1^1 = -P_1^1 - \frac{\Delta_2}{\Delta_1} = 0$$

$$X_2^1 = \frac{1}{2}\left(X_2^1 - Y_2^1 \right) = 0$$

$$R^1(\omega) = X_2^1 - Y_2^1 = R^0(\omega)$$

therefore,

$$X_2^1 = -Y_2^1 = \frac{1}{2} R^0(\omega)$$

Now, we see that the iteration is completed after the first round of calculation and the remainder is no longer the difference between very large numbers, because the remainder is of the same order of magnitude as the corrections X_2^1 and Y_2^1. During the iteration seeking the natural frequencies, we calculate $R^0(\omega_1)$ and $P_i^1(\omega_1)$'s (i=1,2), and increase frequency by an amount $\Delta\omega$ to get $\omega_2 = \omega_1 + \Delta\omega$. Then get $R^0(\omega_2)$ using $P_i^0(\omega_2) = P_i^1(\omega_1)$ (i = 1,2). The above example is for the case r = 3. In other cases for $r > 1$, the spirit is the same (in case r = 1, we cannot use the P-M method).

B. Various ways to get remainder.

There are many ways to calculate a remainder. Each one has a different value at the same frequency ($\omega \neq \omega_k$), but passes through zero at $\omega = \omega_k$.

Some fail and some give good results (see section 2-4-D).

The possible ways arise from the following choices;

1. Which one of the nonvanishing state quantities will be given the value of 1 in the state vector Z_0.

2. Among the r homogeneous equations, as in equation 2-4-5, which r-1 equations will be chosen to get X_1°, X_2°, ...X_{r-1}°.

3. Which of the X_i°'s (i 1,2,...,r-1) in the rth equation is to be regarded as Y° and solved for in terms of the other X°'s.

We will call these different ways by the name "sub-procedures."

C. The relation between a remainder and the frequency determinant.

From the equation 2-4-10, the remainder is

$$R(\omega) = \frac{u_{33}\Delta_3 + u_{32}\Delta_2 + u_{31}\Delta_1}{u_{33}\Delta_1}$$

From equation 2-4-8, we see that Δ's are the minors of an element of the frequency determinant

$$\Delta_1 = M(u_{31}), \quad \Delta_2 = M(u_{32}), \quad \Delta_3 = M(u_{33})$$

where $M(u_{31})$ means the minor of u_{31}.

The remainder is

$$R(\omega) = \frac{\Delta(\omega)}{u_{33}\Delta_1}$$

where $\Delta(\omega)$ is seen to be the frequency determinant.

Expressing a remainder in general in terms of the frequency determinant;

$$R(\omega) = \frac{\Delta(\omega)}{u_{rt}\Delta_{rs}} \qquad (2\text{-}4\text{-}12)$$

u_{rt} : coefficient element corresponding to Y° of rth row.

Δ_{rs} : minor of u_{rs} in $\Delta(\omega)$.

u_{rs} : element in rth row and the column corresponding to state quantity of 1 in Z_0.

D. Failure of the P-M method.

In case the denominator of the remainder is zero, the P-M method certainly fails.* (Refer to equation 2-4-10 and 2-4-12). When the determinant in the denominator becomes nearly equal to zero or changes sign passing through zero, the remainder is difficult to evaluate: at certain frequencies the remainder may show an infinite or jumping phenomenon.

However, this does not necessarily show up at the same frequency with different sub-procedures, because this phenomenon depends solely on the nature of the transfer matrices.

Figure 2-4-1 and Figure 2-4-2 show that while one sub-procedure shows an inifinite phenomenon at certain frequency, another sub-procedure does not show the same phenomenon at this frequency.

Pestel and Mahrenholtz illustrated their method with the case of a homogeneous beam ($2r = 4$). It leads to very good results for the bending frequencies. The reason is, because $r = 2$, there is no phenomenon of indefiniteness causing failure mentioned above and due to purging effect. The P-M method does show improvement also for the case of r greater than two.

For sufficiently high frequencies, the P-M method fails. The reason is as follows, if the columns have already become parallel, purging does not do any good, and if round-off error reaches critical point, where remainder is no longer reliable, this method fails.

*When this is the case, program VIPIPE skips to the next sub-procedure.

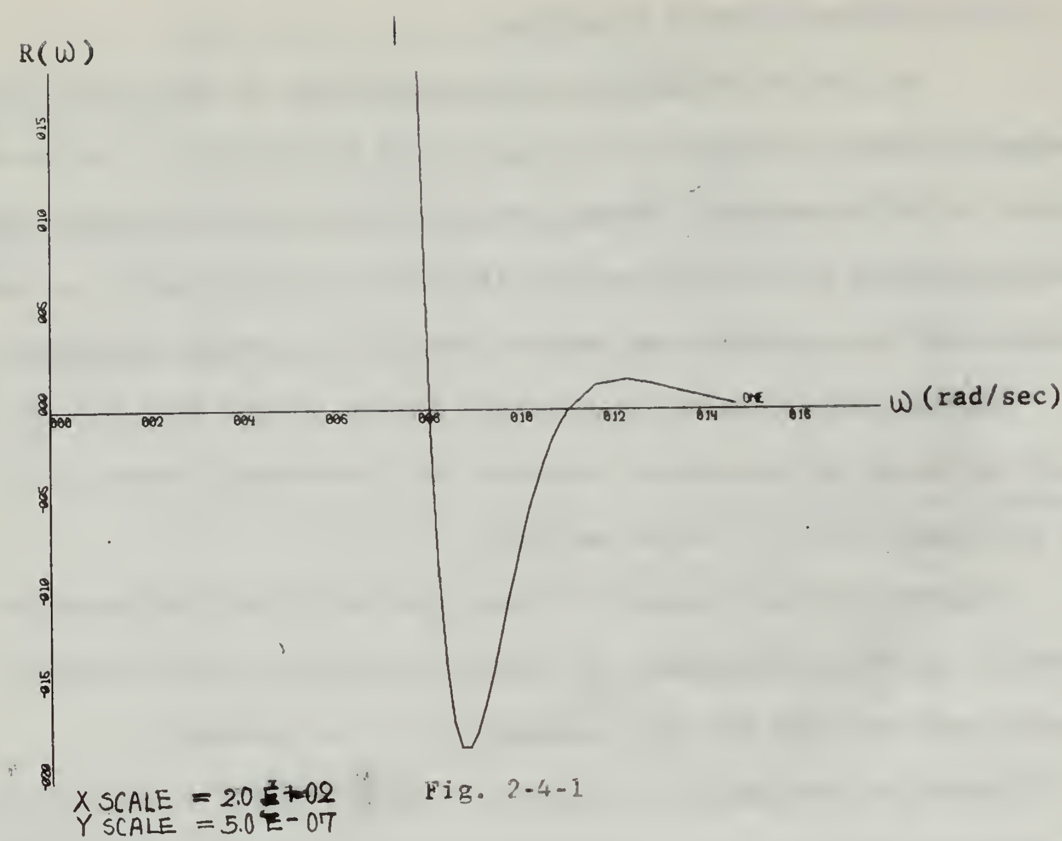


Fig. 2-4-1

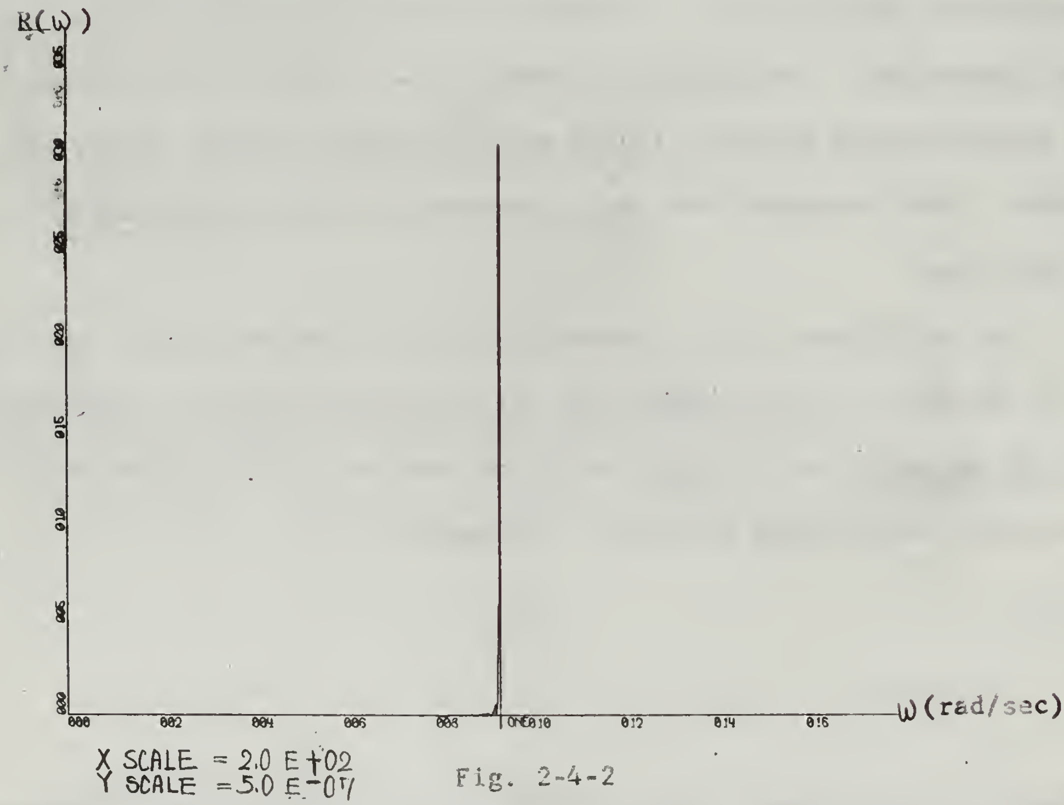


Fig. 2-4-2

Figure 2-4-1 shows fourth and fifth mode frequencies of System 5* at frequency 826.88281950 rad/sec, and 1126.83799669 rad/sec (P-M method-subprocedure A with double precision arithmetic). Figure 2-4-2 shows infinite phenomenon at frequency 939.00298809 rad/sec while Figure 2-4-1 does not show this phenomenon at this Frequency. And Figure 2-4-2 has print out-put with mode frequency 826.88281915, also with 939.00298809 rad/sec (P-M method-subprocedure B with double precision arithmetic). From Figure 2-4-2, we see that the latter is a false mode frequency and the former does not show upon the graph due to the infinite phenomenon of the latter, but we know that there is a mode frequency at 826.88281915 rad/sec (this also can be checked with the frequency and remainder value print output).

2-5 Modified Delta method.

Instead of purging just one column vector among r columns (as in the P-M method) of frequency determinant, Marguerre and Uhrig showed in their paper [2] a possible way of extending the purging concept for problems greater than 4th order.

Through exactly the same method as Delta, we get a frequency determinant. At high frequency, because the columns approach parallelism, the value of the frequency determinant tends toward zero. If the columns are exactly parallel to each other, then of course all methods fail. If they become nearly parallel, then with a finite capacity of handling numbers, the results become meaningless.

Using Maguerre and Uhrig's purging concept, with the help of one column of the frequency determinant, obtain r-1 purging factors to purge

*See Appendix E-3.

another r-1 columns.* For the case of sixth order (r=3) as used in program VIPIPE, the frequency determinant at a certain frequency is:

$$\Delta(\omega) = \begin{vmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{vmatrix}$$

Purging factors got with the help of the third column of $\Delta(\omega)$ are

$$D_1 = \frac{1}{3}\left(\frac{u_{11}}{u_{13}} + \frac{u_{21}}{u_{23}} + \frac{u_{31}}{u_{33}}\right)$$

$$D_2 = \frac{1}{3}\left(\frac{u_{12}}{u_{13}} + \frac{u_{22}}{u_{23}} + \frac{u_{32}}{u_{33}}\right) \qquad (2\text{-}5\text{-}1)$$

Modifying the non-vanishing sub-matrix of the starting state matrix $[Z_0]$, we have

$$[N] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -D_1 & -D_2 & 1 \end{bmatrix} \qquad (2\text{-}5\text{-}2)$$

Then, for the same frequncy, repeat a round of calculation to get the modified frequency determinant.

$$\Delta_m(\omega) = \begin{vmatrix} u_{11} - D_1 u_{13} & u_{12} - D_2 u_{13} & u_{13} \\ u_{21} - D_1 u_{23} & u_{22} - D_2 u_{23} & u_{23} \\ u_{31} - D_1 u_{33} & u_{32} - D_2 u_{33} & u_{33} \end{vmatrix} \qquad (2\text{-}5\text{-}3)$$

From the properties of determinants, $\Delta(\omega)$ is the same as $\Delta_m(\omega)$. But the first and second columns of $\Delta_m(\omega)$ are not the same as those of $\Delta(\omega)$ any more. If the columns of the original frequency determinant were exactly parallel, the modified columns (first and second columns) of the modified frequency determinant will turn out to be all

*We use the term "Modified Delta method" to refer to this method.

zeros. In case the columns were nearly parallel, the first and second columns will turn out to be very small numbers, i.e., we will have purged two columns. So we can expect a more accurate value of the frequency determinant and can go further to higher natural frequencies.

If $D_1 = D_2 = 0$, it degenerates to original frequency determinant, so it does not do any good. If $D_1 = D_2$, it likely means that first and second columns are the same. If this is the case, this method just fails.

In some cases, especially systems of many sections, the frequency determinant does not show parallelism. Experience shows that this method gives good results in most problems. However, this method also fails at sufficiently high frequencies.

2-6 Intermediate conditions - Branched system.

The effect of a joint point can be introduced into the calculation by means of a single point matrix.

This section is mainly for the derivation of the branch point matrix of the Modified Delta method and the Pestel-Mahrenholtz method.

The derivation of branch point matrix for the straight forward transfer method (and thus also for the Delta method) is fully explained in Ref. [3] and [4] , but a brief discussion will be given here for the purpose of refreshing the reader's knowledge and moreover explaining the procedure for the M-D method and the P-M method.

A. Delta method.

Discontinuities are introduced in the forces whereas the main member deflection on each side of joint are the same, and identical to those of the branch at the same joint.

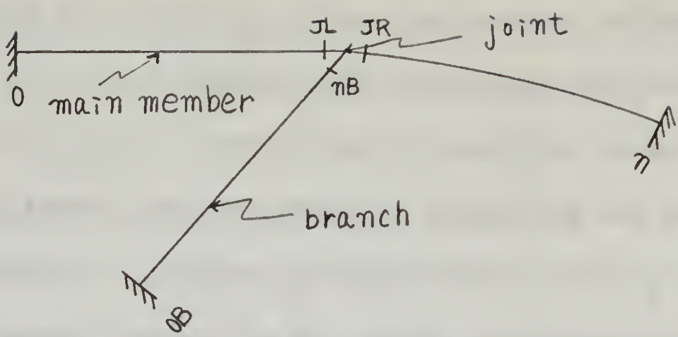


Fig. 2-6-1

By a suitable coordinate transformation, it is possible to express the force system of the branch in terms of the main system coordinates.

A free body diagram for the forces at the joint point is shown below.*

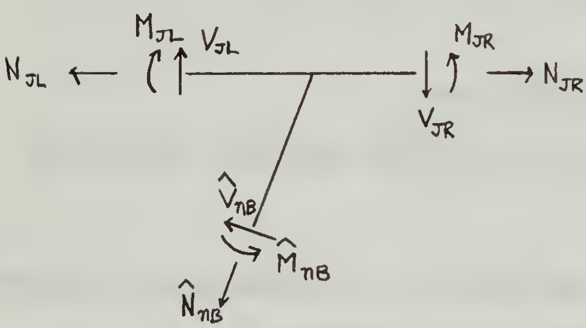


Fig. 2-2-2A

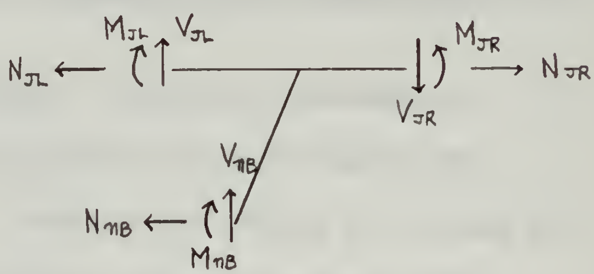


Fig. 2-2-2B

The forces at the joint in matrix form are.

$$\begin{bmatrix} M \\ V \\ N \end{bmatrix}_{JR} = \begin{bmatrix} M \\ V \\ N \end{bmatrix}_{JL} + \begin{bmatrix} M \\ V \\ N \end{bmatrix}_{nB} \qquad (2\text{-}6\text{-}1)$$

The displacement is continuous at the joint, so Z_{JR} can be obtained by simply adding the force components of branch state vector to Z_{JL} .

The column matrix of nonzero component at the starting boundary will be symbolized by V_o. Here $V_o = \{1, 1, 1\}$

*Symbol $\wedge$ indicates that the vector elements are expressed in terms of the branch coordinate system.

$$\hat{Z}_{\pi B} = \begin{bmatrix} \hat{d} \\ \hat{p} \end{bmatrix}_{\pi B} = \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} \cdot V_0 \tag{2-6-2}$$

From equation 2-6-2

$$\hat{P}_{\pi B} = [R_2] \cdot [R_1]^{-1} \cdot \hat{d}_{\pi B} \tag{2-6-3}$$

The matrix product $[R_2] \cdot [R_1]^{-1}$ is called a spring matrix. To transform the coordinate system of the branch into main member coordinates at the joint, we use

$$\begin{aligned} \hat{d}_{\pi B} &= [G_1] \cdot d_{\pi B} \\ P_{\pi B} &= [G_2] \cdot \hat{P}_{\pi B} \end{aligned} \tag{2-6-4}$$

here; $G_1 = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$ $\quad G_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}$

where ϕ is a branch joining angle; refer to Appendix B-5.

then;

$$\begin{aligned} P_{\pi B} &= [G_2] \cdot [R_2] \cdot [R_1] \cdot [G_1] \cdot d_{\pi B} \\ &= [S] \cdot d_{\pi B} \end{aligned} \tag{2-6-5}$$

($[S] = [G_2] \cdot [R_2] \cdot [R_1]^{-1} \cdot [G_1]$; $[S]$ is also called a spring matrix.)

Recalling the relation at the joint

$$d_{JR} = d_{JL} = d_{\pi B}$$

$$p_{JR} = p_{JL} + p_{\pi B} = p_{JL} + [S] \cdot d_{\pi B}$$

we get Z_{JR} in matrix form

$$Z_{JR} = \begin{bmatrix} I & 0 \\ S & I \end{bmatrix} \cdot \begin{bmatrix} d \\ p \end{bmatrix}_{JL}$$

$$\text{or} \qquad = [U_p] \cdot Z_{JL} \tag{2-6-6}$$

Here $[U_p]$ denotes the branch transfer point matrix.

B. Modified Delta method.

(In this section, the use of an asterisk * will denote a purged matrix or vector. Also note that the quantities d and p, with or without * and $\wedge$, represent 3x3 matrices whereas in the preceding subsection they represented 3x1 vectors.)

After we get the purging factors from $\Delta(\omega)$, the new starting matrix at location 0 is

$$[Z_0'] = \begin{bmatrix} 0 \\ N^\circ \end{bmatrix}, \qquad [N^\circ] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -D_1 & -D_2 & 1 \end{bmatrix}$$

At the joint

$$[Z_{JR}^*] = \begin{bmatrix} d^* \\ p^* \end{bmatrix} \qquad [Z_{JL}^*] = \begin{bmatrix} d \\ p \end{bmatrix}_{JL} \cdot [N^\circ]$$

For a branch, let the non-vanishing state matrix of the new starting state matrix $[Z_{0B}]$ be

$$[N_B] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -D_{1B} & -D_{2B} & 1 \end{bmatrix}$$

Here D_{1B}, D_{2B} are branch purging factors, the calculation of which will be explained later. At the joint

$$[\widehat{Z}_{\pi B}^*] = \begin{bmatrix} \hat{d} \\ \hat{p} \end{bmatrix}_{\pi B} \cdot [N_B] = \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} \cdot [N_B] = \begin{bmatrix} R_1^* \\ R_2^* \end{bmatrix}$$

$[R_1]$, $[R_2]$, $[R^*_1]$ & $[R^*_2]$ are defined by this expression.

Just as in equation 2-6-5,

$$
\begin{aligned}
[S^*] &= [G_2]\cdot[R_2^*]\cdot[R_1^*]^{-1}\cdot[G_1] \\
&= [G_2]\cdot[R_2]\cdot[N_B]\cdot[N_B]^{-1}\cdot[R_1]^{-1}\cdot[G_1] \\
&= [G_2]\cdot[R_2]\cdot[R_1]^{-1}[G_1] \\
&= [S]
\end{aligned}
$$

The spring matrix turns out to be the same as in the Delta method.

At the joint

$$
\begin{aligned}
p^*_{JR} &= [P_{JL}]\cdot[N^\circ]\cdot V_0 + [P^*_{\pi B}]\cdot V_0 \\
&= [P^*_{JL}]\cdot V_0 + [S]\cdot[d_{JL}]\cdot V_0
\end{aligned}
$$

$[S]$ is independent of $[N_B]$, so $p^*_{\pi B}$ is independent of $[N_B]$.

For Z_{JR}^* we have

$$
Z_{JR}^* = \begin{bmatrix} I & 0 \\ S & 0 \end{bmatrix} \cdot Z^*_{JL} = [U_p].\ Z^*_{JL}
$$

We see that branch transfer matrix $[U_P]$ is the same as in equation 2-6-6. Purging factors can be obtained from $[R_1]$ or $[R_2]$. The phenomenon of columns approaching parallel could occur in $[R_1]$ and $[R_2]$; VIPIPE computes purging factors from $[R_1]$, because the writer was inclined to think that getting purging factors from $[R_1]$ would do more good in inversion of $[R_1]$. However, $[R_2]$ could have been used.

Purging factors D_{1B} and D_{2B} are obtained in the same way as in equation 2-5-1.

C. Pestel-Mahrenholtz method.

At the joint the equilibrium condition must be satisfied, and the deflections should be continuous.

$$P_{JR}^{*} = P_{JL}^{*} + P_{\Pi B}^{*}$$
$$d_{JR}^{*} = d_{JL}^{*} = d_{\Pi B}^{*}$$

At location 0B (boundary of a branch).

$$\hat{Z}_{0B}^{\circ} = \begin{bmatrix} 0 \\ N_{0B}^{\circ} \end{bmatrix} \cdot X_{0B}^{\circ}$$

Here, $[N_{0B}^{\circ}]$ is the nonvanishing state matrix of $[\hat{Z}_{0B}^{\circ}]$ and X_{0B}° is a branch correction vector,

let

$$[N_{0B}^{\circ}] = \begin{bmatrix} P_{0B}^{\circ} & 0 & 0 \\ P_{1B}^{\circ} & 1 & 0 \\ P_{2B}^{\circ} & 0 & 1 \end{bmatrix} \qquad X_{0B} = \begin{bmatrix} X_{0B}^{\circ} \\ X_{1B}^{\circ} \\ X_{2B}^{\circ} \end{bmatrix}$$

Then, the state vector at location 0B for the built-in, in-plane case is

$$\hat{Z}_{0B}^{\circ} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ P_{0B}^{\circ} X_{0B}^{\circ} \\ P_{1B}^{\circ} X_{0B}^{\circ} \quad X_{1B}^{\circ} \\ P_{2B}^{\circ} X_{0B}^{\circ} \quad X_{2B}^{\circ} \end{bmatrix} \qquad (2\text{-}6\text{-}7)$$

At the joint

$$Z_{JL}^{*} = \begin{bmatrix} d \\ P \end{bmatrix}_{JL} \cdot [N^{\circ}] \cdot X^{\circ} \qquad Z_{JR}^{*} = \begin{bmatrix} d^{*} \\ P^{*} \end{bmatrix}_{JR} \cdot X^{\circ}$$

$$\hat{Z}_{\Pi B}^{*} = \begin{bmatrix} \hat{d} \\ \hat{P} \end{bmatrix} \cdot [N_{0B}^{\circ}] \cdot X_{0B}^{\circ} = \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} \cdot [N_{0B}^{\circ}] \cdot X_{0B}^{\circ} = \begin{bmatrix} R_1^{*} \\ R_2^{*} \end{bmatrix} \cdot X_{0B}^{\circ}$$

where $[R_1^*]$ and $[R_2^*]$ are

$$[R_1^*] = [R_1]\cdot[N^\circ_{0B}] \qquad [R_2^*] = [R_2]\cdot[N^\circ_{0B}]$$

For $[N^\circ]$ and $[X^\circ]$, refer to section 2-4. Also $[R_1]$, $[R_2]$ are expressed in the same way as the preceding sub-section B.

$$\begin{aligned}
[S^*] &= [G_2]\cdot[R_2^*]\cdot[R_1^*]^{-1}\cdot[G_1] \\
&= [G_2]\cdot[R_2]\cdot[N^\circ_{0B}]\cdot[N^\circ_{0B}]^{-1}\cdot[R_1]^{-1}\cdot[G_1] \\
&= [G_2]\cdot[R_2]\cdot[R_1]^{-1}\cdot[G_1] \\
&= [S]
\end{aligned}$$

The spring matrix also turns out to be the same as in the Delta method.

From the requirement for continuous deflections at the joint we get

$$\begin{aligned}
d^*_{\pi B} &= d^*_{JR} = d^*_{JL} \\
d^*_{\pi B} &= [G_1]^{-1}\cdot[R_1]\cdot[N^\circ_{0B}]\cdot X^\circ_{0B} \\
&= [d_{JL}]\cdot[N^\circ]\cdot X^\circ \qquad (2\text{-}6\text{-}8)
\end{aligned}$$

and

$$\begin{aligned}
p^*_{\pi B} &= [S]\cdot d^*_{\pi B} = [S]\cdot[d_{JL}]\cdot[N^\circ]\cdot X^\circ \\
p^*_{JR} &= [p^*_{JL}]\cdot X^\circ + [S]\cdot[d^*_{JL}]\cdot X^\circ
\end{aligned}$$

For Z^*_{JR} we have

$$Z^*_{JR} = \begin{bmatrix} I & 0 \\ S & I \end{bmatrix}\cdot Z^*_{JL} = [U_P]\, Z^*_{JL}$$

We see that the branch transfer matrix $[U_P]$ is also the same as in equation 2-6-6. From equation 2-6-8, the branch correction vector is

$$X^\circ_{0B} = [R_1^*]\cdot[G_1]\cdot[d^*_{JL}]\cdot X^\circ \qquad (2\text{-}6\text{-}9)$$

After we get the main member correction vector (X°), we can get branch state correction factors in terms of X° values. As explained in section 2-4, iteration is performed in the same way.

2-7 Computer.

In FORTRAN 63, floating point quantities have an exponent and fractional part. In single precision arithmetic, the fractional part has 36 bits and the precision is approximately 10 decimal digits. In double precision arithmetic, the fractional part has 85 bits and the precision is approximately 25 decimal digits. In both cases the range of numbers is from 10^{-308} to 10^{308}.

In the FORTRAN 63 version of VIPIPE (as described later), either single precision (for conservation of computer time) or double precision (for greater computational accuracy) can be selected. In the FORTRAN 60 version, only single precision is available.

The natural frequencies of a planar piping system are found using program VIPIPE. Program VIPIPE has been tested with FORTRAN 60 and 63 using digital computer CDC 1604. The number of natural frequencies that may be found using VIPIPE and the accuracy, especially at high frequency, depends upon the floating point significant figure capacity. And the significant figure capacity required for any system is a function of the characteristics of the system itself.

CHAPTER III

DISCUSSION

3-1 Accuracy of methods and assurance of the solution.

Confidence in accuracy of methods was established by comparing frequencies of several systems for which these frequencies could be obtained by other means, i.e., by means of either a closed form analytical solution or by recourse to the Ref. [5] .

The accuracy is very good for the lower natural frequencies in the values got using the Delta method, the Modified Delta method, and the Pestel-Mahrenholtz's method.

In most cases, up to fourth mode frequencies, the difference between the system natural frequencies obtained by analytical means and those obtained using VIPIPE are less than 0.004%. As frequency increases we got more accurate natural frequency values from the M-D method and the P-M method comparing to the original transfer method (Delta method).

In the sample problems reported in Appendix E, we have found that for straight, simple configurations (where the phenomenon of the columns becoming parallel is pronounced) the P-M and M-D methods give one or two more natural frequencies than can be obtained by the Delta method or the original transfer method. However, in more complicated configurations it seems to be possible to get three or four more natural frequencies. In the simpler cases, there are comparison values available from "exact" theory so that the accuracy of the results may be established. However, in the more complicated cases, such comparison values are not available. However, we have reasonable assurance of the integrity

of these results since we have been able to get the same values from the M-D method and from several different P-M subprocedures. Thus, using these methods, we seem to be able to double or triple the frequency range in which valid results may be obtained.

Details of the systems analyzed and the results obtained may be found in Appendix E.

3-2 Limitations and some suggestions.

Non-stiff hangers may be introduced into the system by a point matrix with appreciable linear and/or torsional spring compliances. But, for stiff hangers the transfer method fails, so we cannot use program VIPIPE for natural frequencies of a system which has stiff hangers. In particular, the suggestion of Fink [4] that rigid hangers can be successfully handled by use of idealized very stiff exterior springs proves to be untenable.

Marguerre and Uhrig suggested possible way to overcome the difficulty relating to stiff hangers by introducing unknowns [2] . Introducing unknowns means method abandoning the transfer concept, so we cannot use VIPIPE directly to solve such a problem. However, the writer thinks that there is a possibility of modifying VIPIPE to solve such problems by getting state quantities at each node and assuming values for the unknowns. For details refer to reference [2] .

BIBLIOGRAPHY

1. Pestel, E., and O. Mahrenholtz: Zum numerischen Problem der Eigenwertbestimmung mit Übertragungsmatrizen, Ingr.-Arch., vol. 28, pp. 255-262, 1959.

2. Marguerre, K., and R. Uhrig: Das Üebertragungsverfahren und seine Grenzen, Z. angew. Math. u. Mech., Vol. 43, 1963.

3. Pestel, E. C., and F. A. Leckie: Matrix Methods in Elastomechanics, McGraw-Hill Book Company, 1963.

4. Fink, George E.: Vibration Analysis of Piping Systems. Masters Degree Thesis, United States Naval Postgraduate School, 1964.

5. den Hartog, J. P.: Mechanical Vibrations, 4th Ed., McGraw-Hill Book Company, 1956.

6. Frazer, R. A., W. J. Duncan, and A. R. Collar: Elementary Matrices, Cambridge Univ. Press., 1952.

7. Brock, J. E.: Shear Distrubution in Piping. Heating Piping and Air Conditioning, Vol. 35, pp. 141-143, 1963.

APPENDIX A

DESCRIPTION OF PROGRAM

A-1 General remarks

Program VIPIPE is a CDC FORTRAN 63 language digital computer program designed for use in the vibration analysis of planar piping systems, originally developed by George E. Fink [4] . The program VIPIPE is modified and augmented by the writer to use Delta method (same results as original transfer method, but with less amount of calculation), Modified Delta method, and Pestel and Mahrenholtz's remainder method with double precision or single precision arithmetic.

In- and out-of-plane undamped natural frequencies may be found through the use of program VIPIPE. How this program may be used is fully explained in Appendix B. Considerable flexibility with regard to the mathematical model employed is provided in the program. For instance, one may use combination of a distributed mass treatment for straight section with a lumped mass treatment for curved sections (this method gives the greatest accuracy coupled with least machine time) or a lumped mass for all sections. Also one may consider or neglect the effect of shear deflection and rotational inertia, singly or together.

Codes for the boundary conditions and various methods are interpreted within the program.

Problem solutions proceed by iterative processes with radian frequency the independent variable. The sign of the frequency determinant or remainder is used to control the search for and convergence to a system natural frequency.

Output and transfer to the next method/or transfer to the next problem or to the end is provided in the program when it reaches significant figure limitation or previously set iteration limit, and also when a method fails.

All subroutines are provided in binary deck in single precision and double precision arithmetic in the FORTRAN 63 version. Modifications of the program may be accomplished by changing FORTRAN cards only in the main program; the subroutines are in binary to conserve space and time.

A-2 Program structure

The program consists of a six section main body and twenty-seven subroutines. In the FORTRAN 63 library, GRAPH subroutine is not available. So GRAPH subroutine is provided in binary, and built into the program. Also, in the FORTRAN 63 library, hyperbolic functions are not available, so these functions are computed in exponential form.

The function of each section of the main body and of each subroutine is given below.

A. Main body sections and their functions.

INPUT	Controls read-in of data, allocates storage locations for arrays and subscripted variables, decides the first method to perform.
INVARIANTS	Computes and assigns appropriate subscripts to such system component characteristics as moment of inertia and radius of gyration, computes frequency increment for the first two modes, and sets the solution acceptability criterion.

CONTROL IN-PLANE	Constructs transfer matrix of the system for the in-plane case by calling for various subroutines in a sequence governed by the system geometry, the mathematical model specified, and the desired assumptions.
CONTROL OUT-OF-PLANE	Performs the same function as CONTROL IN-PLANE but for the out-of-plane case.
ITERATION	Evaluates the frequency determinant (modified frequency determinant in the M-D method) or the remainder and utilizes its sign in controlling the iterative search for mode frequencies. Initiates output when the specified frequency range has been fully explored, the required number of modes found, the specified iteration number has been reached, or the significant figure limit of computer reached.
OUTPUT	Controls output format, provides graph output, seeks the method to perform.

B. Subroutines and their functions.

SUBSEC	Subdivides components to be treated by lumping mass on the basis of L/D ratio and starting frequency or required mode numbers. (L/D ratio for curved section means ratio of radius of curvature to diameter and included angle of arc, as applicable.)
MATMUL	Constructs the transfer matrix of those components treated by lumping mass for only one point mass and massless subsection.
FINMAT	Constructs the system state matrix (6x3) by successive premultiplication of the partial system state matrix

	by the transfer matrix of each component as soon as each of the latter is constructed. For the lumped mass system, gives successive premultiplication by the transfer matrix constructed by MATMUL as many times as the number of lumped masses.
FINBRA	Constructs the state matrix of a branch system by successive pre-multiplication of the partial branch state matrix by the transfer matrix of each component of a branch as soon as each of the latter is constructed. For the lumped mass system, gives successive pre-multiplication by the transfer matrix constructed by MATMUL as many times as the number of lumped masses.
STATEM	Constructs starting state matrix from the starting boundary condition of main member, for each method.
STATEB	Constructs starting state matrix of a branch system from the far end boundary condition of a branch for each method. Computes the purging factors of the branch in the M-D method.
BRCOR	Constructs branch correction matrix for the assumed branch starting state matrix at a certain frequency in the P-M method. Constructs branch purging matrix in the M-D method.
DELMA	Constructs frequency determinant in the Delta method. Evaluates the purging factors from the frequency determinant of the Delta method and constructs the modified frequency determinant after another round of calculation in the M-D method. For the P-M

method, computes the correction factors for each sub-procedure. For each sub-procedure, if it fails by the fact that denominator is zero, the rows are changed and a second test is made. It also provides transferring to the next method or sub-procedure if it fails.

DISTM, DISTMO, SFIELD, SFIELO, CFIELD, CFIELO, POINT, POINO, RIGID, RIGIO, STIFCO, HANGER, HANGEO, BRANCH, BRANCO, STAVEC, STAVEO, INVERT.

For the above subroutines refer to reference [4] .

A-3 Program nomenclature.

B1,B2,B3	Correction factors for assumed state quantities of the far end of a branch.
BC	Discriminant. Controls boundary correction in the P-M and M-D method.
BL	Number of the last iteration.
D1,D2	Purging factors in the M-D method. Correction factors in the P-M method.
DEL	Array of numerical values of frequency determinants and array of remainders.
DV	Numerical value of a determinant in denominator of X1 and X2.
EDL	Array of numerical values of frequency determinants or remainders in single precision for input to GRAPH.
ERM	A discriminant same as REM in the problem statement data card.
FF	Discriminant. Controls output to provide a print

output that analysis is terminanted due to significant figure limitation in the P-M method.

FINK	Discriminant. Controls constructing the frequency determinant or modified frequency determinant in the M-D method.
HH	Discriminant. Controls transferring to another sub-procedure or to another way of testing in case one fails in the P-M method.
HMH	Discriminant. Controls subsectioning in lumped mass system.
K	Discriminant. Controls grid in the GRAPH output.
KA	Discriminant. Terminates iteration when specified iteration number is finished.
KKK	Discriminant. Controls transferring to the output in case the last sub-procedure in the P-M method fails or controls transferring to another three sub-procedures (D,E,F) after three sub-procedures (A,B,C) are finished in case subprocedure C fails in Selection 3 (for Selection, see Appendix B-9-6).
MAT	Discriminant. Controls matrix multiplication in the lumped mass system.
NUMPTS	Number of points in one graph (except the last).
NUMEND	Number of points in the last graph.
P1, P2	Assumed starting state quantities of the main member.
PMM	Discriminant. Controls three methods (Delta, M-D, P-M method) specified.

PPM	Discriminant. Assures that same procedures are used for out-of-plane solution that were used for in-plane solution.
PM	Discriminant. Controls three methods to be used.
PP	Discriminant. Controls the three sub-procedures (A,B,C) in the P-M method.
RR	Discriminant. Controls testing by interchanging two rows in case one sub-procedure fails because the denominator of X1 or/and X2 is zero in the P-M method.
R3	Array name. Matrix product $[R_1]^{-1} \cdot [G_1]$, used in the branch correction.
REM	Discriminant. Controls three ways of selecting sub-procedures in the P-M method.
Q1,Q2,Q3	Assumed starting state quantities of a branch.
SS	Discriminant. Controls the three sub-procedures (D, E,F) in the P-M method.
XX	Array name. Branch correction matrix (3x3) in the P-M method and branch purging matrix in the M-D method.
X1,X2,Y1,Y2	Correction factors for assumed starting state quantities of the main member in the P-M method.
YY	Array name. Branch correction matrix (3x3x23).

For other nomenclature see reference [4] .

APPENDIX B

INSTRUCTION FOR PROGRAM USE

B-1 General remarks.

Appendix B is intended to provide complete instructions for using program VIPIPE. VIPIPE is FORTRAN 63 language program with twenty-seven subroutines. All subroutines are provided in a binary deck so that the compiling time is reduced as is the volume of the deck.

Program VIPIPE can be used in single precision or double precision arithmetic. Also VIPIPE is provided in FORTRAN 60 language with single precision.

The Delta method, the M-D method, and the P-M method are all incorporated in program VIPIPE. The M-D method and the P-M method do appear to permit obtaining more natural frequencies (one,two, or possibly several more) than can be obtained with the Delta method (same result as the original transfer method). However, the P-M method has the disadvantage of introducing an odd behavior of the remainder. In some cases, this odd behavior can easily be misinterpreted so as to lead to false frequency determination. However, there are a whole group of associated P-M modifications* and the odd behaviors occur at different frequencies for the different modifications. Thus, by making the calculation more than once, using a different modification each time, the true natural frequencies can be sorted out.

Program VIPIPE uses the Delta method as the basic procedure. If more frequencies are required than have been obtained by this method,

*P-M modifications means sub-procedures of the P-M method.

one may then use one or more of the available associated P-M modifications and/or the M-D method. Graphical output will assist in identifying the natural frequencies. The writer recommends to use the P-M modifications as original method in getting more natural frequencies, and to use the M-D method for purposes of comparison.

The accuracy and number of natural frequencies that can be obtained using the methods depends on the capacity of handling numbers accurately. As we can see in Appendix E, double precision arithmetic gives better results especially for complicated systems for which many matrix multiplication are performed. But using double precision arithmetic results in considerable increase of machine time. And also for various methods, the machine time is different. The Delta method and the P-M method each take about the same amount of machine time, but the M-D method takes almost twice this time.

So a user should consider the frequency range under investigation and the nature or characteristics of the system as well as the machine time.

For comparison purposes Section B-9-6 of this Appendix lists machine times for one iteration for some example problems using each of three methods with double and single precision arithmetic.

Complicated systems are divided into one main member and branches. System components are defined as sections of pipe, straight or of constant curvature, an elbow, a spring hanger, a coupling, a flange. Diameters, wall thickness, density, elastic and shear modulus may vary from component to component, but may not vary within a component. Both torsional and linear spring rates (for hangers) must have finite values. Thus perfectly rigid hangers cannot be treated.

B-2 Coordinate system.

A local coordinate system is constructed at each point of interest in the system. A right handed coordinate system is used whose X axis is the local tangent to the central axis of the main member, always directed away from the left end. The X-Y plane coincides with the quiescent plane of system. The Z axis is always directed downward. The direction of the local Y axis is chosen so as to complete the right handed system.

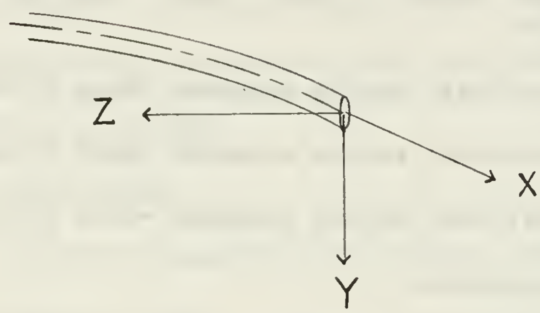


Fig. B-2-1

The branch coordinate starts from the end of the branch farthest from the joint point with the main member. The plane of the branch coincides with the X-Y plane. The positive Z axis of the branch and main member are parallel and have the same sense. The Y axis is oriented so as to complete a right handed system.

B-3 Data required.

Extensive properties.

a. outside diameter (D) - inches.

b. wall thickness (DI) - inches

c. radius of curvature (RHO) - inches (Negative, if center of curvature lies on negative Y axis).

d. length (TL) - inches (for straight section only; for curved sections, length is computed by the program).

e. arc central angle (THETA) - degrees

f. linear spring constant in X direction (CLX) - pounds per inch

g. linear spring constant in Y direction (CLY) - pounds per inch

h. linear spring constant in Z direction (CLZ) - pounds per inch

i. torsional spring constant about X axis (CTX) - in-lb/radian

j. torsional spring constant about Y axis (CTY) - in-lb/radian

k. torsional spring constant about Z axis (CTZ) - in-lb/radian

Intensive properties

a. density of component (AAMU) - lb/ft^3

b. elastic modulus (AE) - psi

c. shear modulus (AG) - psi

B-4 Component identification.*

Components are identified by two numbers.

A. All components have a component sequence number starting from the most left component of the main member. This sequence number is identified by the order of data cards containing component extensive and intensive properties. This can be understood from the figure B-4-1.

*Aslo see pp. B-5 to B-7 of reference [4] .

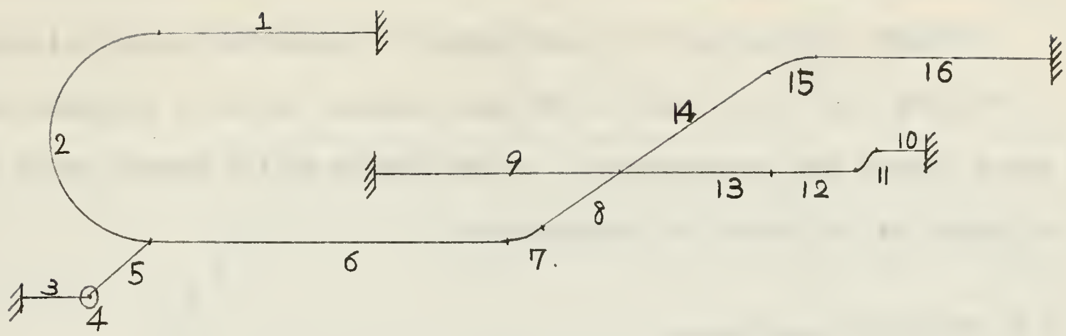


Fig. B-4-1

B. A branch component identification number (NNBR) is read in as data. It indicates whether it is in a branch or main member, if it is in branch, the position in branch.

NNBR 0	main member
NNBR 1	the first component of a branch (farthest from the intersection)
NNBR 2	intermediate component of a branch
NNBR 3	the last component of a branch
NNBR -1	the only component of a branch

This can be understood from the figure B-4-2

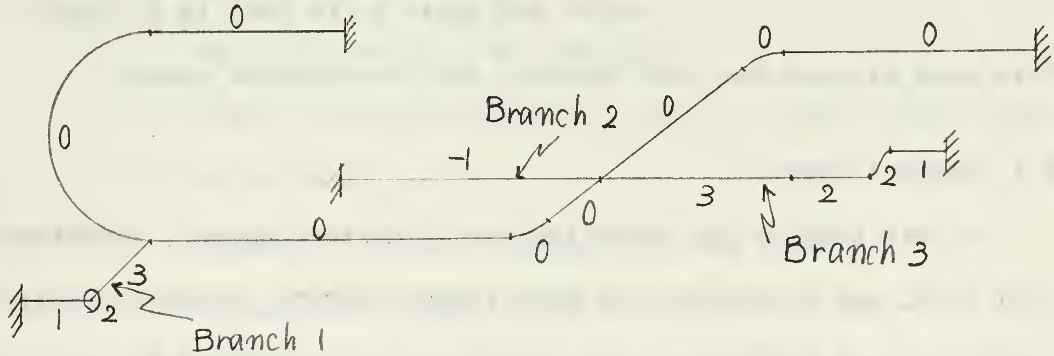


Fig. B-4-2

B-5 Branch joining angle

Branch joining angle to main member is measured counter-clockwise in degrees from the X axis of the main member, as it is oriented immediately before the inter-section, to the X axis of the branch as it is oriented at the point of intersection.

B-6 Boundary conditions*

Five common boundary conditions, those of a fixed, free, pinned, propped, or a roller supported end are indicated by the code given. The program interprets the code and forms the required state matrix (6x3).

Boundary condition code:

a. SV 1 fixed end

b. SV 2 free end

c. SV 3 pinned end

d. SV 4 propped end

e. SV 5 roller supported end

f. SV 0 other than one of the above. (The user must construct the necessary state vector and cause to be read in as data).

This code is used for both in-plane and out-of-plane cases.

B-7 Control cards.

A user sets up for execution with a master control, a FORTRAN control card, and combination of END, FINIS, EXECUTE, transfer cards, binary cards, and FORTRAN cards properly placed in FORTRAN 63 deck (JOB and END cards are used in FORTRAN 60). Figure B-7-1 shows the deck assembly

*For further details, see pp. B-8 to B-10 of reference [4].

for FORTRAN 63 use.

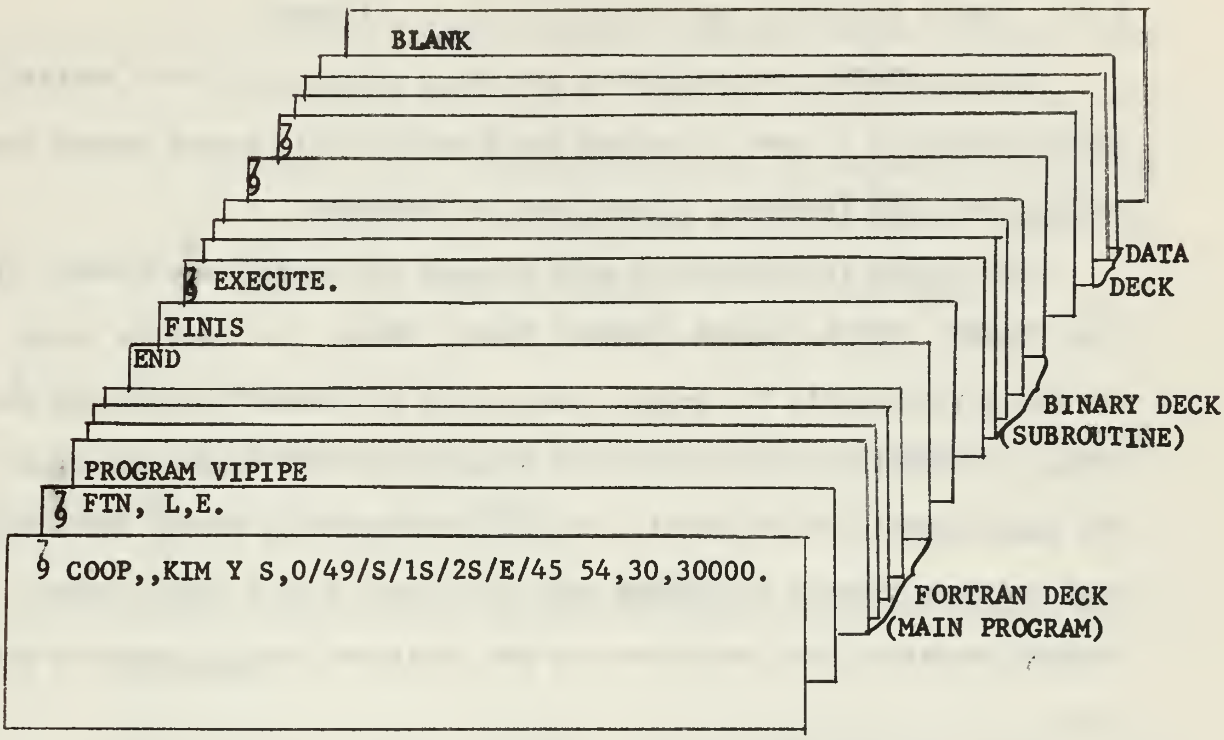


Fig. B-7-1

B-8 Data deck assembly.

The data deck is ordered as follows:

a. problem statement card.

b. boundary condition data card(s).

c. branch joining angle data card(s). (deleted when system has no branches)

d. component extensive properties data card(s).

e. component intensive properties data card(s).

The user may obtain solutions for any number of problems desired; in this case each problem data deck is assembled just behind the other.

Following the last problem deck a blank card should be added.

B-9 Flexibility of program VIPIPE.

B-9-1 Double precision and single precision arithmetic.

Program VIPIPE can use double precision process only with FORTRAN 63. Double precision is used in system state matrix (UU), branch system state matrix (VV), and frequency determinant or remainder.

Type double is declared in main program and subroutines FINMAT, FINBRA, STATEM, STATEB, BRANCH, BRANCO, BRCOR, DELMA. By removing cards declaring type double the program can easily be changed into single precision. Subroutines are provided in binary deck with single precision and double precision arithmetic, so with corresponding binary subroutine deck and main program in FORTRAN with or without a card (card number 000120) declaring type double we can use single or double precision arithmetic.

B-9-2 GRAPH.

A user can get GRAPH output by option. The values of the frequency determinant and the values of the remainder are not usually of the same order of magnitude throughout the frequency range. Moreover, because of the infinite phenomenon of the P-M method, it is not generally useful or convenient to get one graph for the whole frequency range.

1. Number of points in one graph can be changed by changing NUMPTS in card 006090.
2. Can get graph with grid or without grid by changing card 006100.

   K = 1 with grid

   K = 0 without grid.

3. Graphs are labeled from 1,2,3,...up to 10. Graphs beyond tenth one will be labeled as 10. The last graph is labeled LAST.

4. For each graph, the whole frequency range is moved (except the first one) so that the first point of each graph starts from 0. The corresponding value of frequency determinant (or remainder) for the first point and NUMPTS (NUMEND for the last graph) are printed in the print output.

B-9-3 Shear area form factor.

A shear area form factor of one half is built in program.* This may be changed by changing card 000600.

B-9-4 Starting frequency, frequency increment and solution acceptability criterion.

Using the M-D or the P-M method, we are expecting to get more natural frequencies at high frequency. Thus choosing the proper starting frequency for the M-D or the P-M method, following the Delta method can save machine time. This can be done by putting the wanted starting frequency in problem statement data card.

To compute the starting frequency increment, the program constructs a synthetic straight pipe, equal in length to the length of the main member, and having diameter, wall thickness, and intensive properties equal in magnitude to a weighted average of those properties of the composite members. One quarter of the fundamental frequency of this synthetic pipe is taken as the starting frequency increment (card 001390). And one thousandth of this increment is taken acceptability criterion (card 001410). When the iteration reaches this solution acceptability

*See reference [7] .

criterion, the corresponding natural frequency is determined using linear interpolation.

This frequency increment is for the first two mode frequencies, but the acceptability criterion is the same for all mode frequencies. After second mode frequency is found, frequency increment is one tenth of the difference between the last two mode frequencies found (card 003950).

These can be altered by changing cards. The writer found that a smaller frequency increment gives better results especially at high frequency with the P-M method.

B-9-5 Subsectioning.

Subsectioning for the lumped mass treatment of components is carried on in subroutine SUBSEC. A component is divided into maximum of twelve subsections.

For all components having a length to diameter ratio greater than six, the number of subsections is computed as two times the value of a parameter HM which appears in SUBSEC (HM is number of modes to be sought). If HM is greater than six, HM is given the value of six. In case the starting frequency is greater than 800 radians per second, HM is also given the value of six. HM can be controlled by a parameter HMH in the main program (card 001250). If HMH is zero, HMH does not affect the value of HM. By changing HMH to other than zero we can change the value of HM equal to that of HMH.

Poor subsectioning might give some loss of accuracy in determining natural frequencies, especially at high frequency. Checking the natural frequencies by increasing the number of subsections using parameter HMH is recommended.

B-9-6 Machine time and printing line limit.

In FORTRAN 63, estimated time is set to thirty minutes and printing line limit is set at 30000 lines in VIPIPE. These limits can be altered by changing the master control card. If a limit is reached, computation stops automatically.

For reference, machine time for one iteration of some example problems is given below.

	Example System*	Delta meth.	M-D meth.	P-M meth.
Double Prec.	system 1	0.131 sec.	0.247 sec.	0.144 sec.
	system 4	2.10 sec.	4.22 sec.	2.14 sec.
	system 5	4.799 sec.	6.90 sec.	3.505 sec.
Single Prec.	system 1	0.044 sec.	0.117 sec.	0.045 sec.
	system 5	0.377 sec.	0.968 sec.	0.49 sec.

B-9-7 Performing various methods.

One selects the methods to be employed by use of the problem statement data card. Then the program picks the first method in the INPUT section of the main program with a discriminant PM, in the order: Delta method, M-D method, P-M method.

Finishing a method, the OUTPUT section of the main program opens to the next method. Finishing all specified methods, the program goes to the out-of-plane portion, to the next problem, or to the end.

In the P-M method, there are six sub-procedures. These sub-procedures are controlled by discriminatns PP, SS.

PP : 0.	subprocedure A,	SS : 0.	sub-procedure D,
PP : 1.	sub-procedure B,	SS : 1.	sub-procedure E,
PP : 2.	sub-procedure C,	SS : 2.	sub-procedure F.

*For example systems refer to Appendix E.

Each sub-procedure has another possible way by interchanging rows of final determinant (refer to section 2-4). If one fails, the alternative one is tested. If one succeeds, then go to the next sub-procedure. There are three ways of selecting sub-procedures by discriminant REM.

REM: 2. three sub-procedures in the order: A,B,C. - Selection 1

REM: 1. three sub-procedures in the order: D,E,F. - Selection 2

REM: 0. six sub-procedures in the order: A,B,C,D,E,F. - Selection 3

If one wants just one sub-procedure, by changing PP to a value of 2 (card 000710) in selection 1 only sub-procedure C is used, or changing SS to a value of 2 (card 001000) in selection 2 only sub-procedure F is used. If one wants just two sub-procedures, by changing PP or SS to a value of 1 sub-procedure B and C or sub-procedure E and F, respectively, are used.

B-9-8 Limiting the number of iterations.

One can limit the number of iterations, for each problem, by a parameter HL in the problem statement data card. In the P-M method, because of the infinite phenomenon, the frequency increment in iteration can become very small. Thus the iteration might keep going on indefinitely.

The program permits six hundred iterations as maximum. The writer found three hundred iterations to be adequate to get nine natural frequencies with the acceptability criterion set in the program.

B-9-9 End conditions.

A boundary condition state vector must have three non-zero and three zero elements. Frequently, however, an actual condition may not meet this specification. For instance, an end may be supported or connected to a piece of machinery. Then we approximate it as being held in a

mounting block having specified linear and torsional spring rates, and to this attach a fictious massless short section.

Through the use of this sort of fictious end, a proper end condition state vector may be constructed for any real situation with little or no loss of accuracy.

For hangers exhibiting coupling, systems having unusual flexibility, and flanges and couplings refer to pp. B-19 to B-20 of reference [4] .

B-10 Format of data

B-10-1 Problem statement

The card has 15 fields under control of the following field specifications.

312, 4F3.0, 2F8.0, 4F2.0, 2F4.0

Field names and the information each field conveys are given below in order.

A. Field names and content.

1.	NS	number of components.
2.	NBR	number of branches.
3.	IOP	code for in-and/or out-of-plane solutions.
4.	HM	number of modes sought.
5.	AMYK	code calling for the use of lumped or distributed mass model
6.	SD	code calling for shear deflection inclusion or neglect
7.	RI	code calling for rotational inertia effect inclusion or neglect
8.	OMEGAG	upper limit of frequency range of investigation
9.	OMEGAI	lower limit of frequency range of investigation and initial iteration frequency. If 0 is placed, it will begin at .01 radians per sec.

10. GDABC code calling for print output.

11. BRC code calling for the inclusion or exclusion of branch correction factors in P-M method and purging factors in M-D method.

12. GRAPH code specifying whether there is to be graph output.

13. REM code for choosing methods in P-M method.

14. PMM code calling for the combination of 3 methods of Delta, M-D, P-M method which a user wants.

15. HL number of iterations.

B. Codes used.

1. IOP 0 Both in-and out-of-plane mode frequency sought (in this case, in-plane frequencies are sought first).

2. IOP 1 In-plane frequencies are sought.

3. IOP 2 Out-of-plane frequencies sought.

4. AMYK 1. Lumped mass model.

5. AMYK 0. Distributed mass model.

6. SD 0. Consider shear deflection.

7. SD 1. Neglect shear deflection.

8. RI 0. Consider rotational inertia.

9. RI 1. Neglect rotational inertia.

10. GDABC 0. Print solution only.

11. GDABC 1. Print solution and input data.

12. GDABC 2. Print solution and input data and graph data.

13. BRC 0. Do not consider branch correction factors or purging factors.

14. BRC 1. Consider the branch correction factors or purging factors.

15. GRAPH 0. No graph output.

16. GRAPH 1. Graph output.

17. REM 0. All six sub-procedures in the P-M method (A,B, C,D,E,F).

18. REM 1. Three sub-procedures in the P-M method (D,E,F).

19. REM 2. Three sub-procedures in the P-M method (A,B,C).

20. PMM 1. Delta method only.

21. PMM 10. M-D method only.

22. PMM 100. P-M method only.

23. PMM 11. Delta, M-D method.

24. PMM 101. Delta, P-M method.

25. PMM 110. M-D, P-M method.

26. PMM 111. Delta, M-D, P-M method.

B-10-2 Boundary conditions.

Field specification 22F7.3

Boundary conditions are read in separately for in- and out-of-plane cases. When both cases are to be investigated, in-plane case data are read in first.

N + 2 fields must be filled in (N is number of branches). The first and last fields are first and last boundary condition codes of the main member. Branch boundary conditions are entered in the intermediate fields in accordance with the branch sequence number. When a boundary condition is not describable by one of the codes, a zero must be entered in its data field. For each zero an additional data card is required. The limitation in non-zero state quantitites at the boundary must be three (refer to B-9-9).

B-10-3 Branch joining angle (degrees).

Field specification is: 20F7.3

B-10-4 Component extensive properties and branch component identification.

A data card for each system component, ordered by sequence number.

Field specification is: 3F5.2, 2F6.2, 6F7.2, I2

Field names are: in order,

D, DI, RHO, TL, THETA, CLX, CLY, CLZ, CTX, CTY, CTZ, NNBR

B-10-5 Intensive properties.

The first intensive properties data card has four fields with specification:

F7.3, 2E10.2, F2.0

Field names are: AAMU, AE, AG, DD*

*When the intensive properties of all components are the same, the DD field is left blank. When the intensive properties are different, the DD field is non-zero, then an intensive data card must be prepared for each of the remaining components (including hangers). Additional data cards are identical to the first one except that the DD field is blank.

APPENDIX C

FLOW DIAGRAMS

C-1 General remarks.

Flow diagrams are included for all parts of the main program, and for several subroutines. Since CONTROL OUT OF PLANE, FINBRA flow diagrams are identical in structure to those of CONTROL IN PLANE and FINMAT respectively, only the latter are included. Reader can refer to the flow diagram of STAVEC and STAVEO of reference [4]. Since the flow diagram structures of SUBSEC, BRANCH and BRANCO are similar to those of reference [4], they are not included here.

Flow diagrams of other subroutines that are not included in this paper are relatively simple ones, so the reader can refer directly to the program.

C-2 Index of flow diagram.

Main program.

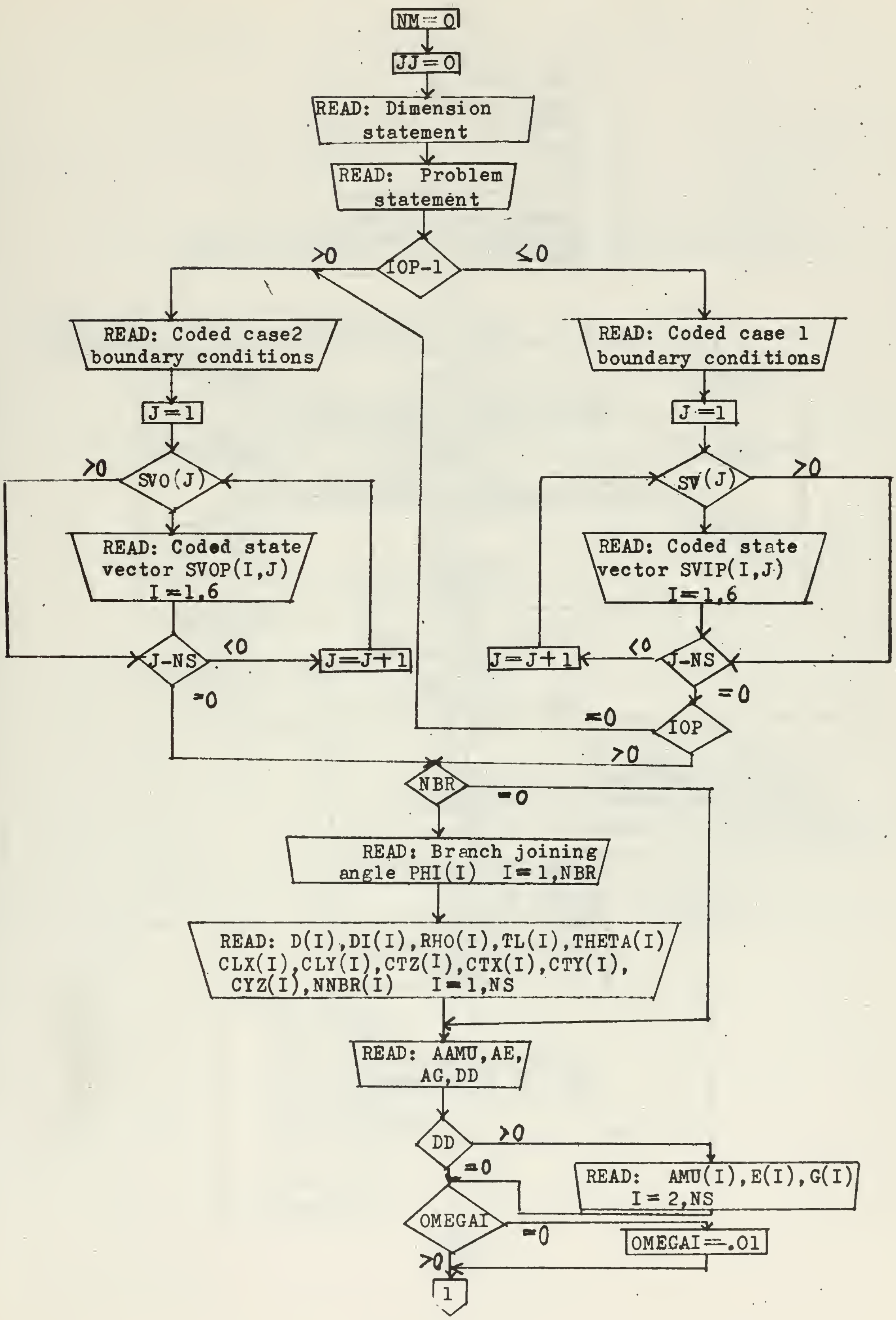
Flow Diagram.	Page.
INPUT	63
INVARIANTS	65
CONTROL IN PLANE	66
ITERATION	68
OUTPUT	70

Subroutines

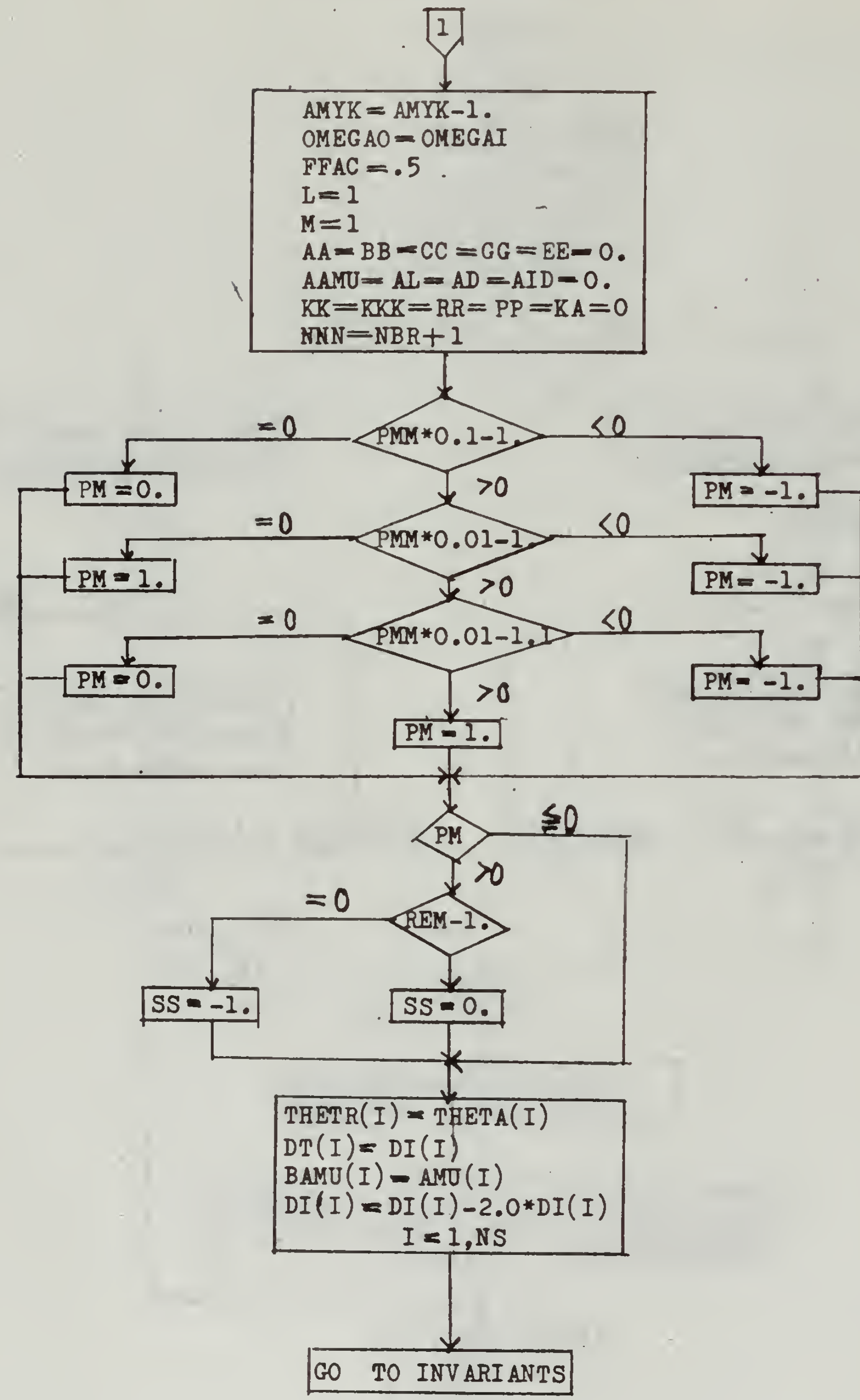
FINMAT	73
STATEM	74
STATEB	75
BRCOR	76
DELMA	77

C-3 Flow Diagrams

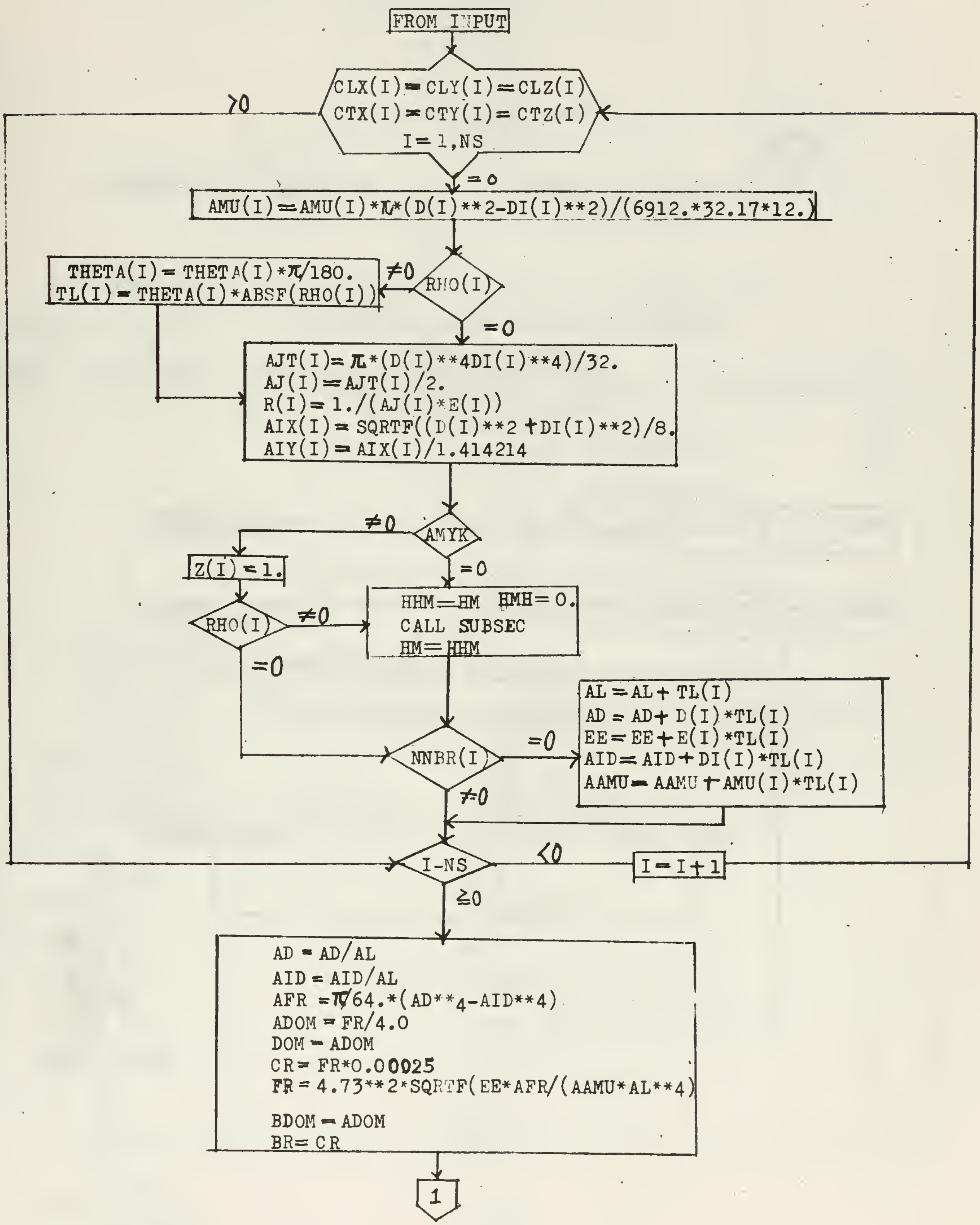
INPUT FLOW DIAGRAM



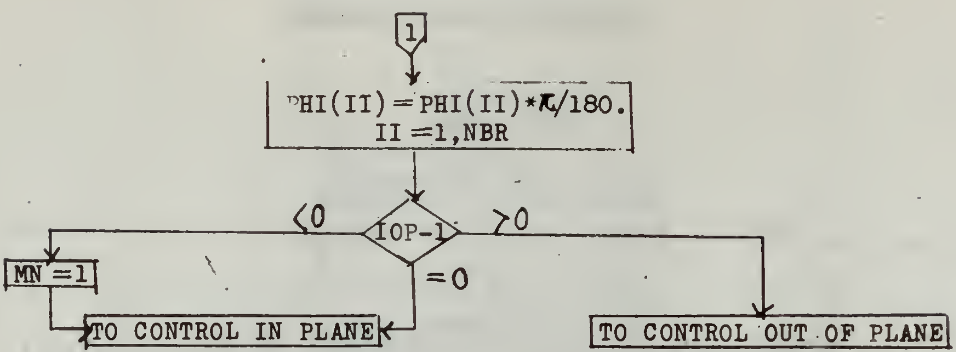
INPUT FLOW DIAGRAM(CONT.)



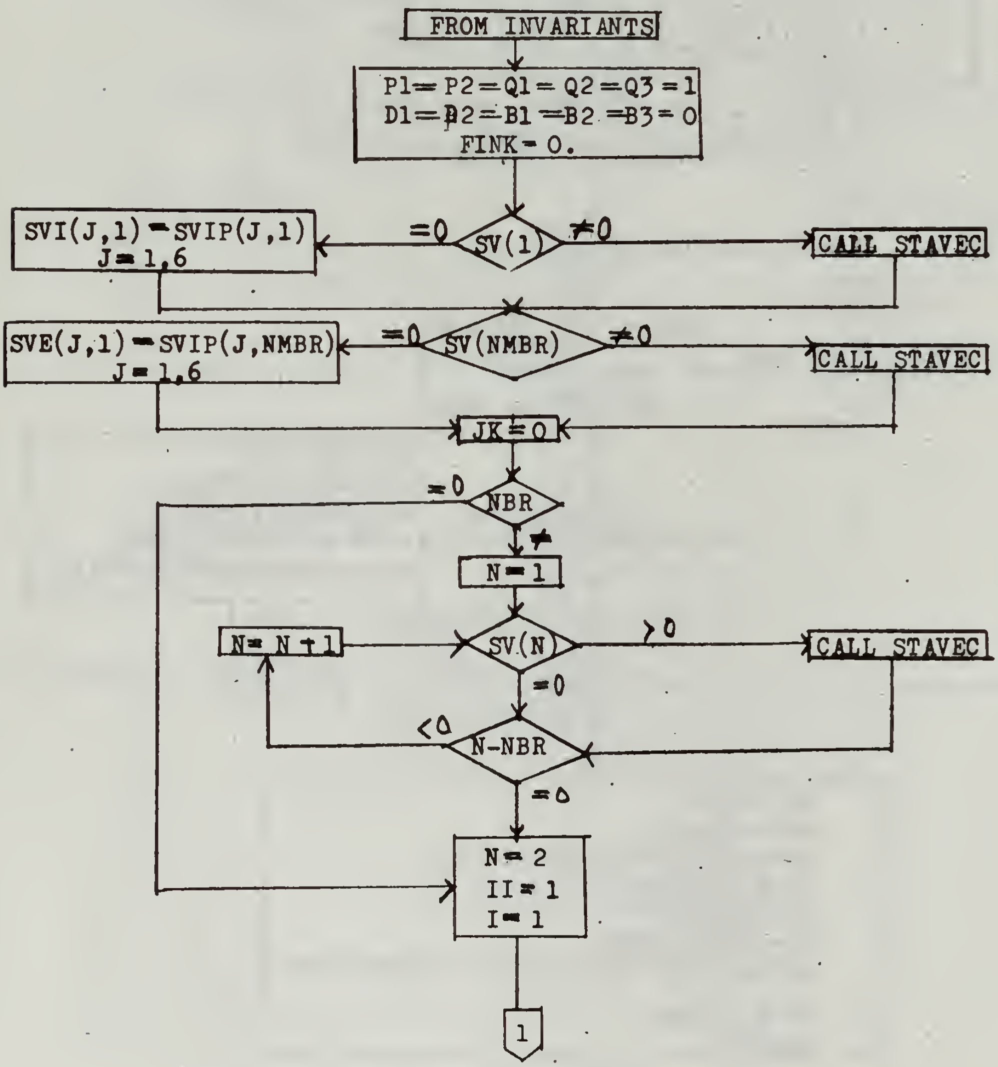
INVARIANTS FLOW DIAGRAM



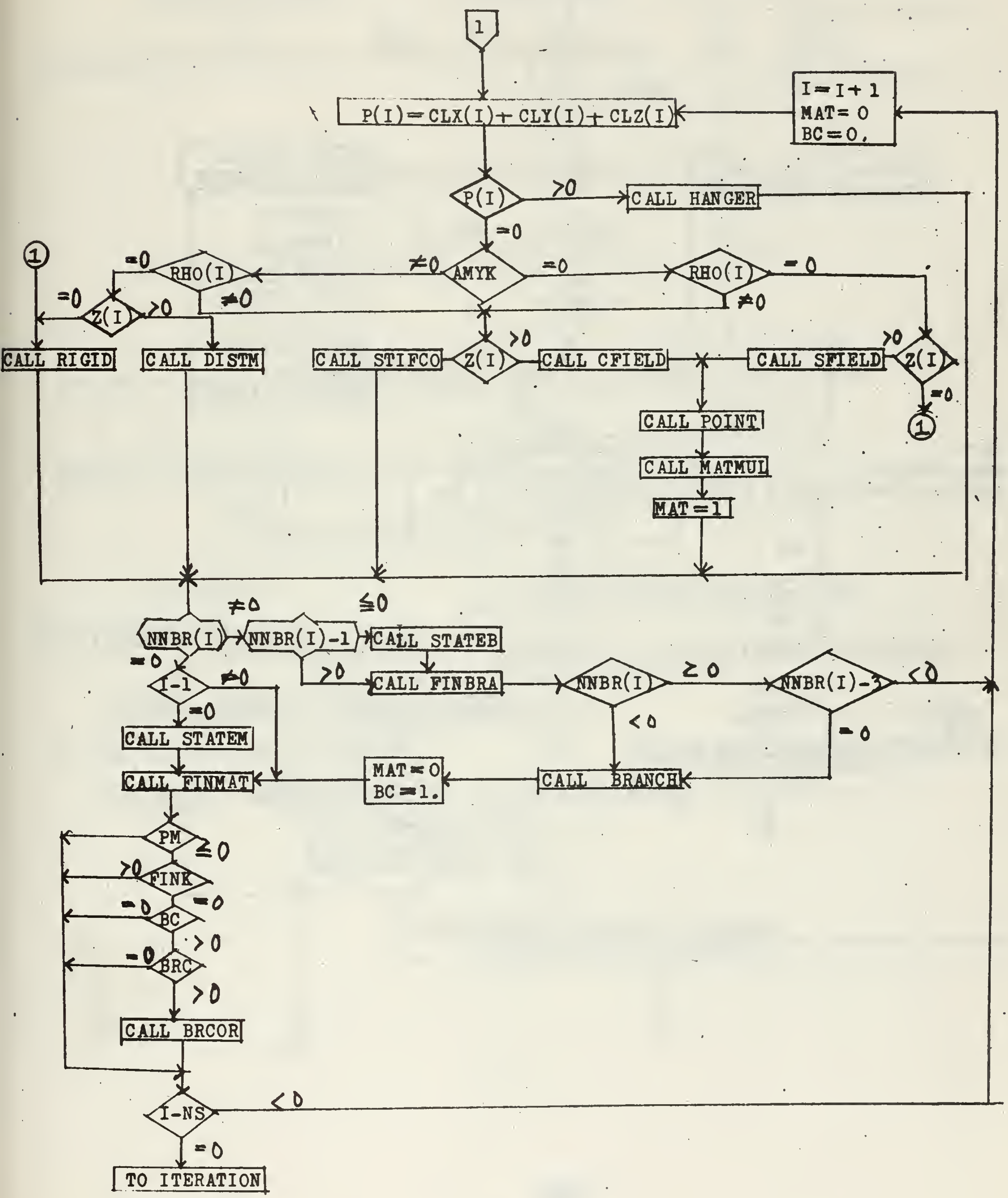
INVARIANTS FLOW DIAGRAM(CONT.)



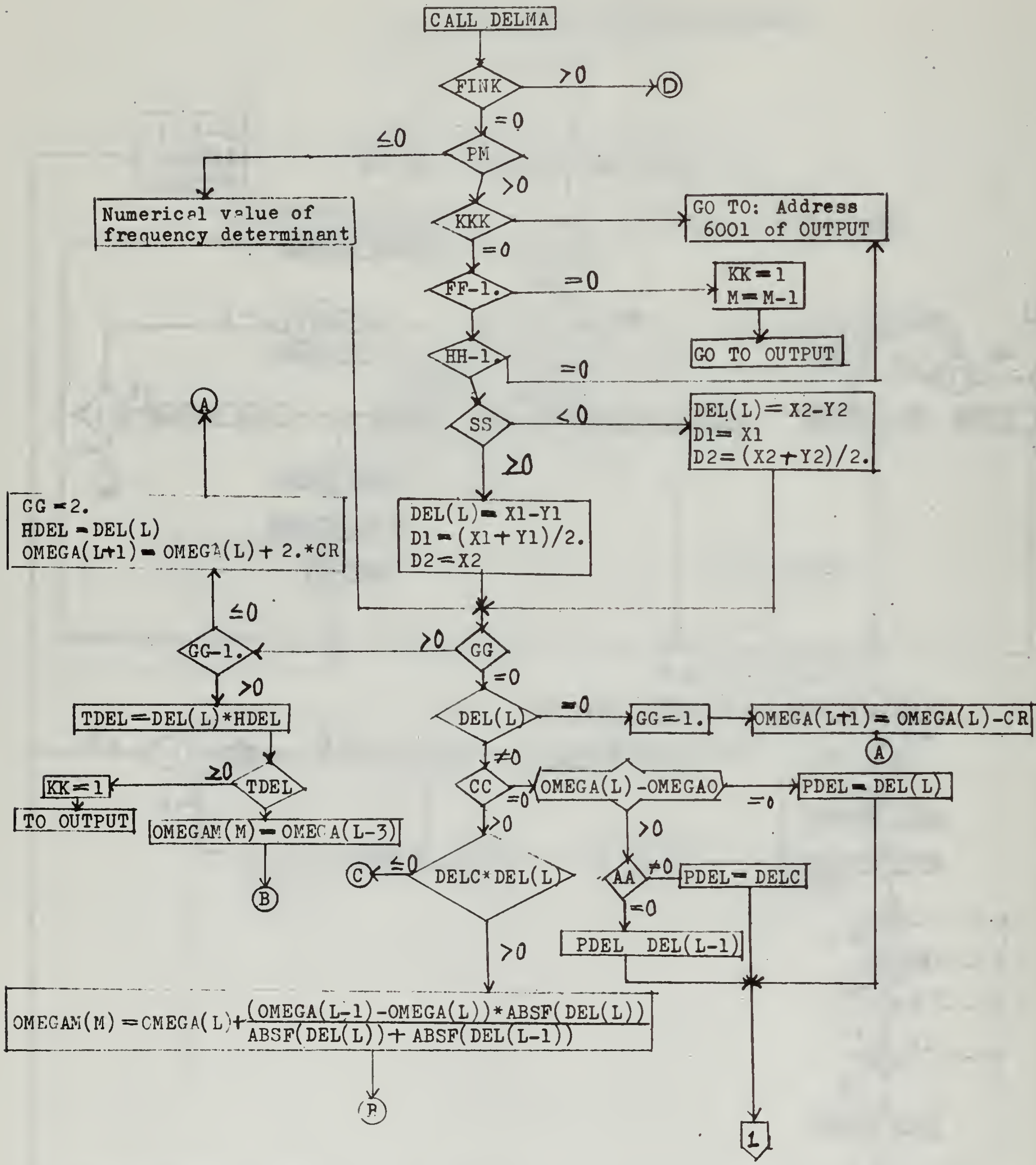
CONTROL IN PLANE FLOW DIAGRAM



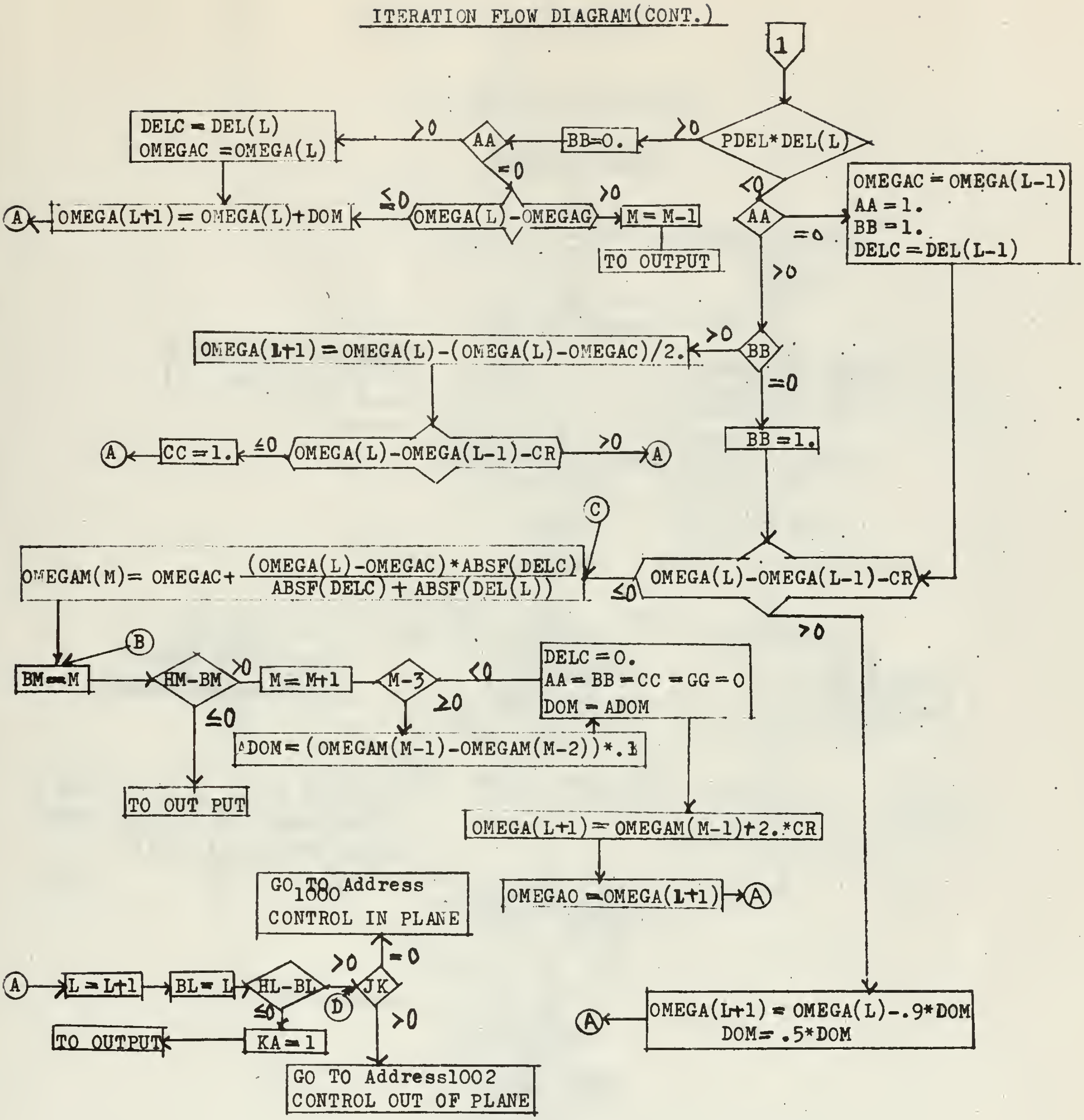
CONTROL IN PLANE DIAGRAM (CONT.)



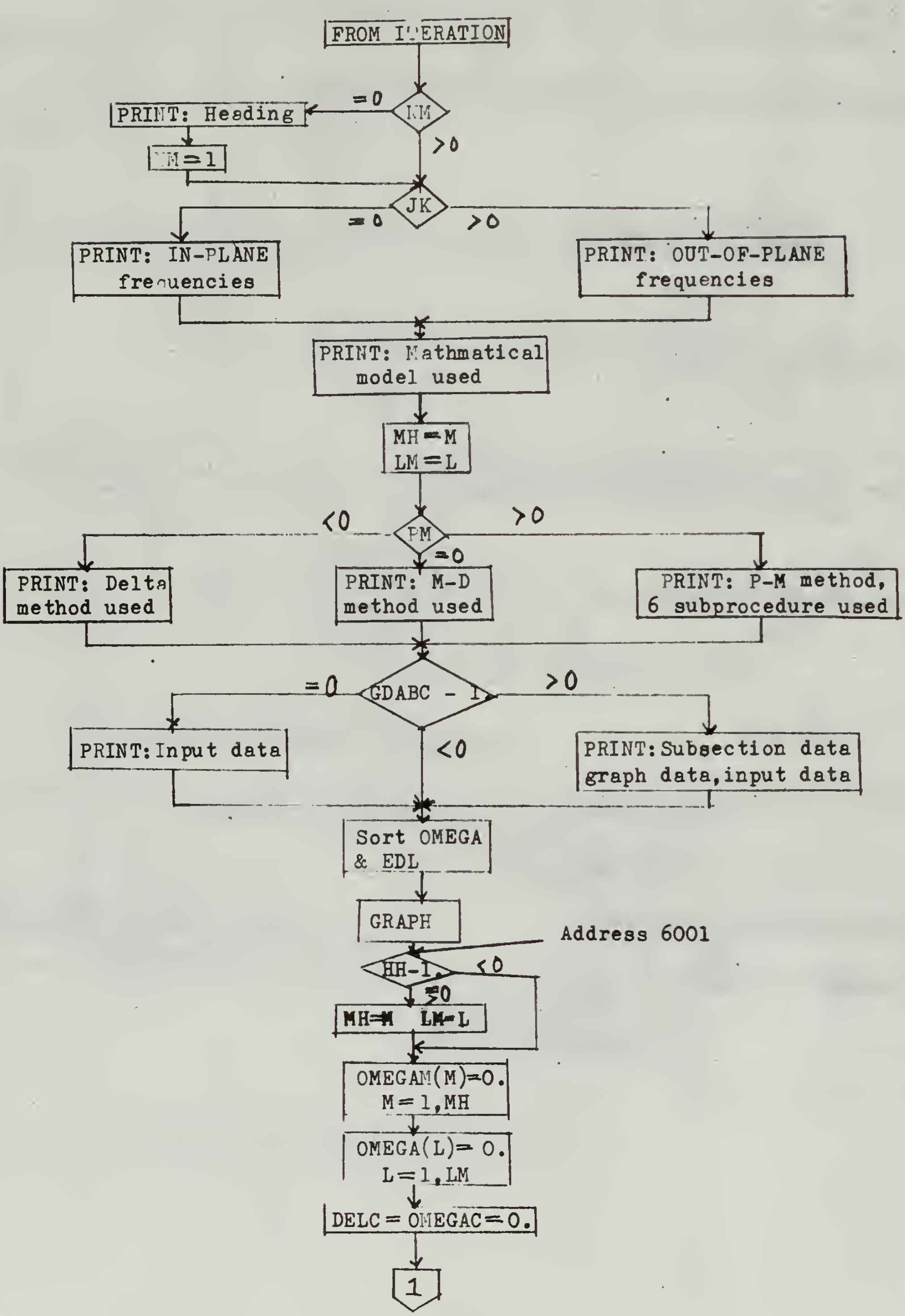
ITERATION FLOW DIAGRAM



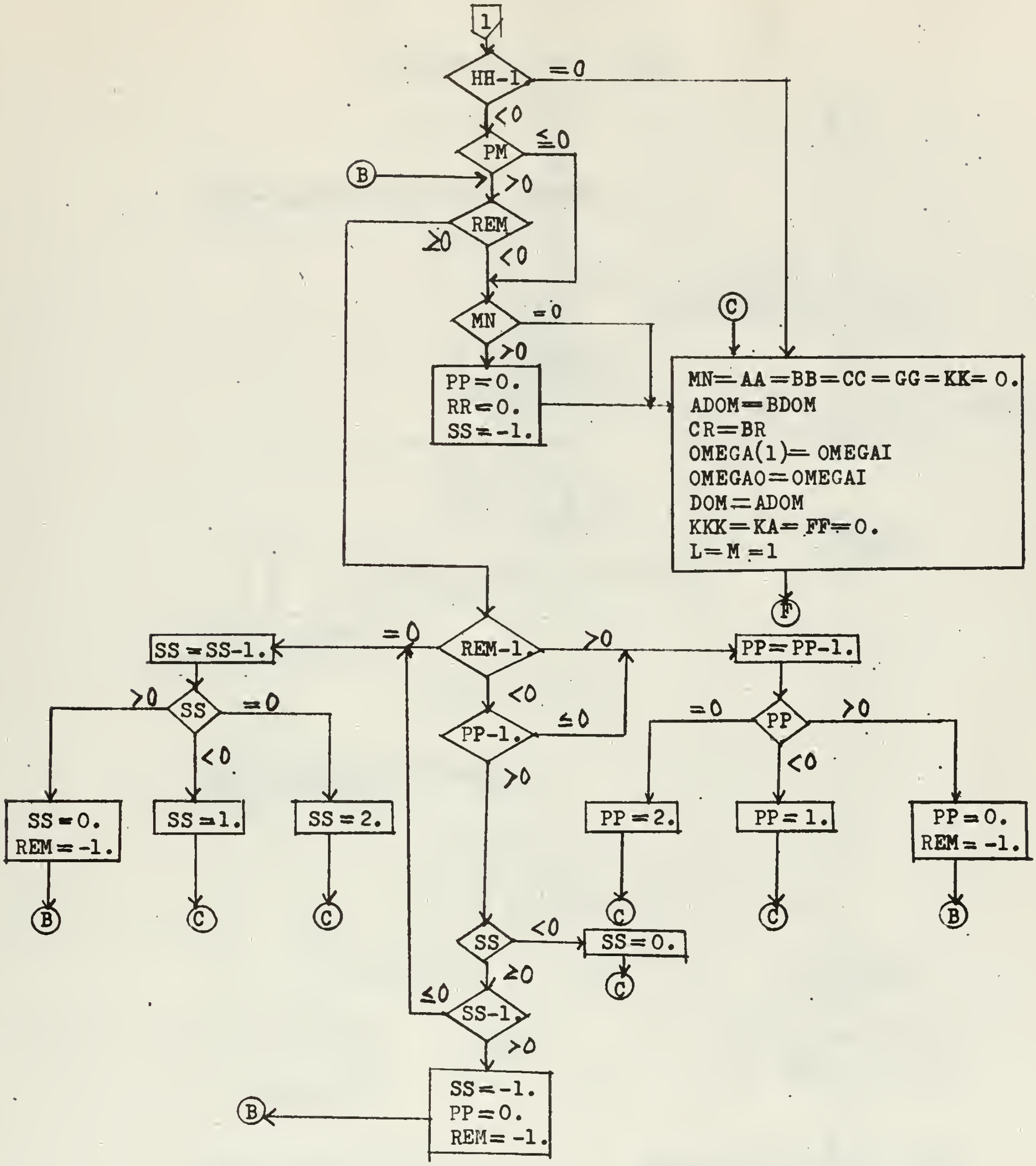
ITERATION FLOW DIAGRAM(CONT.)

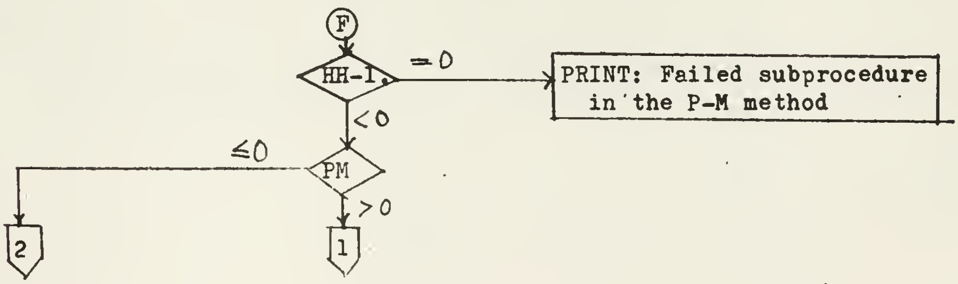


OUTPUT FLOW DIAGRAM

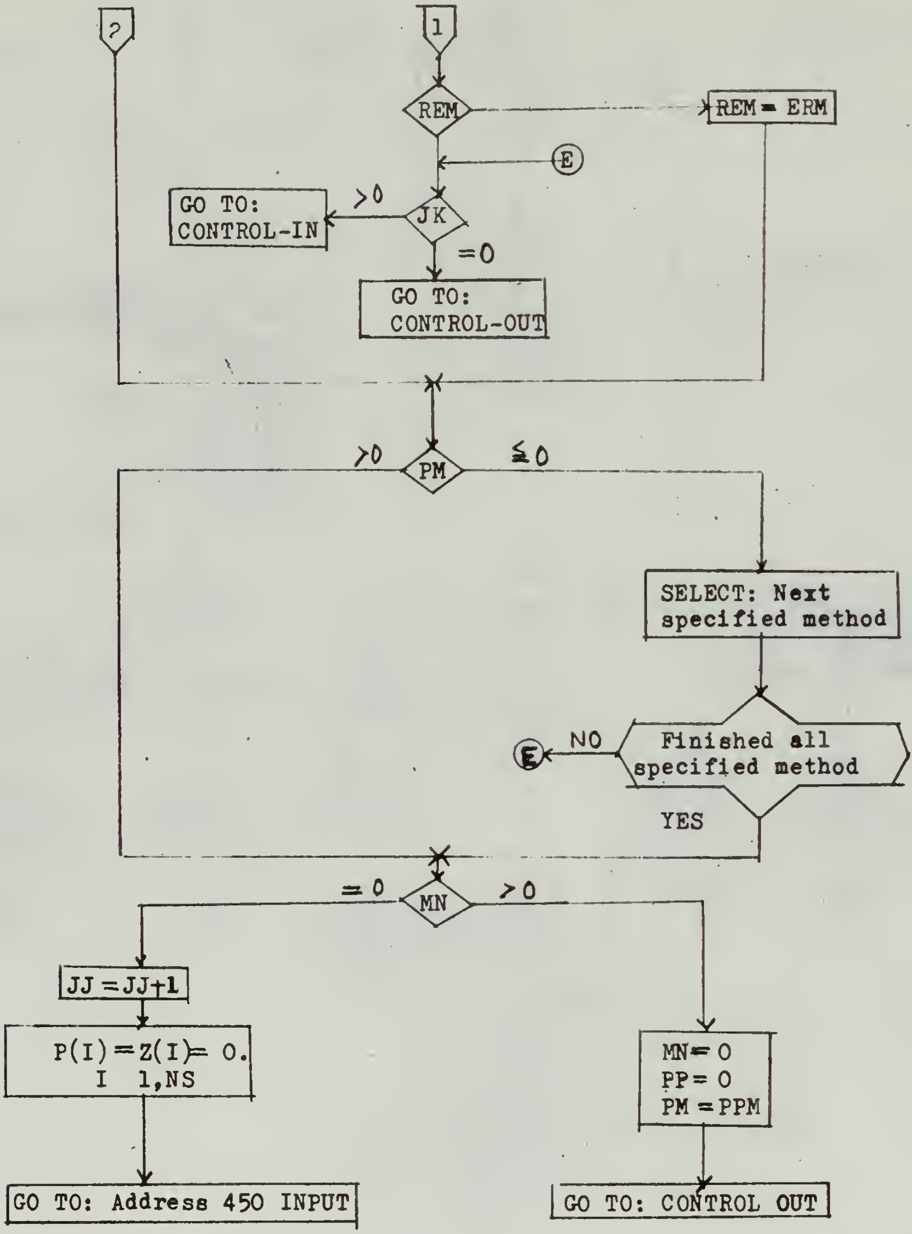


OUTPUT FLOW DIAGRAM(CONT.)

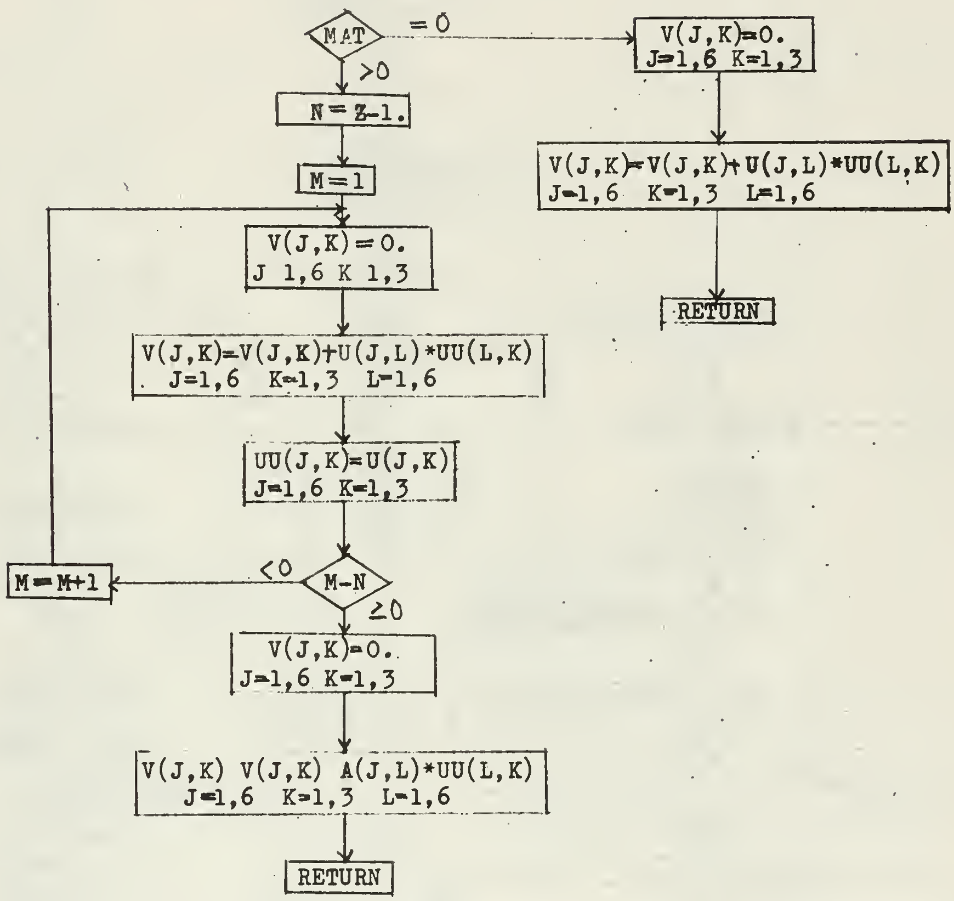




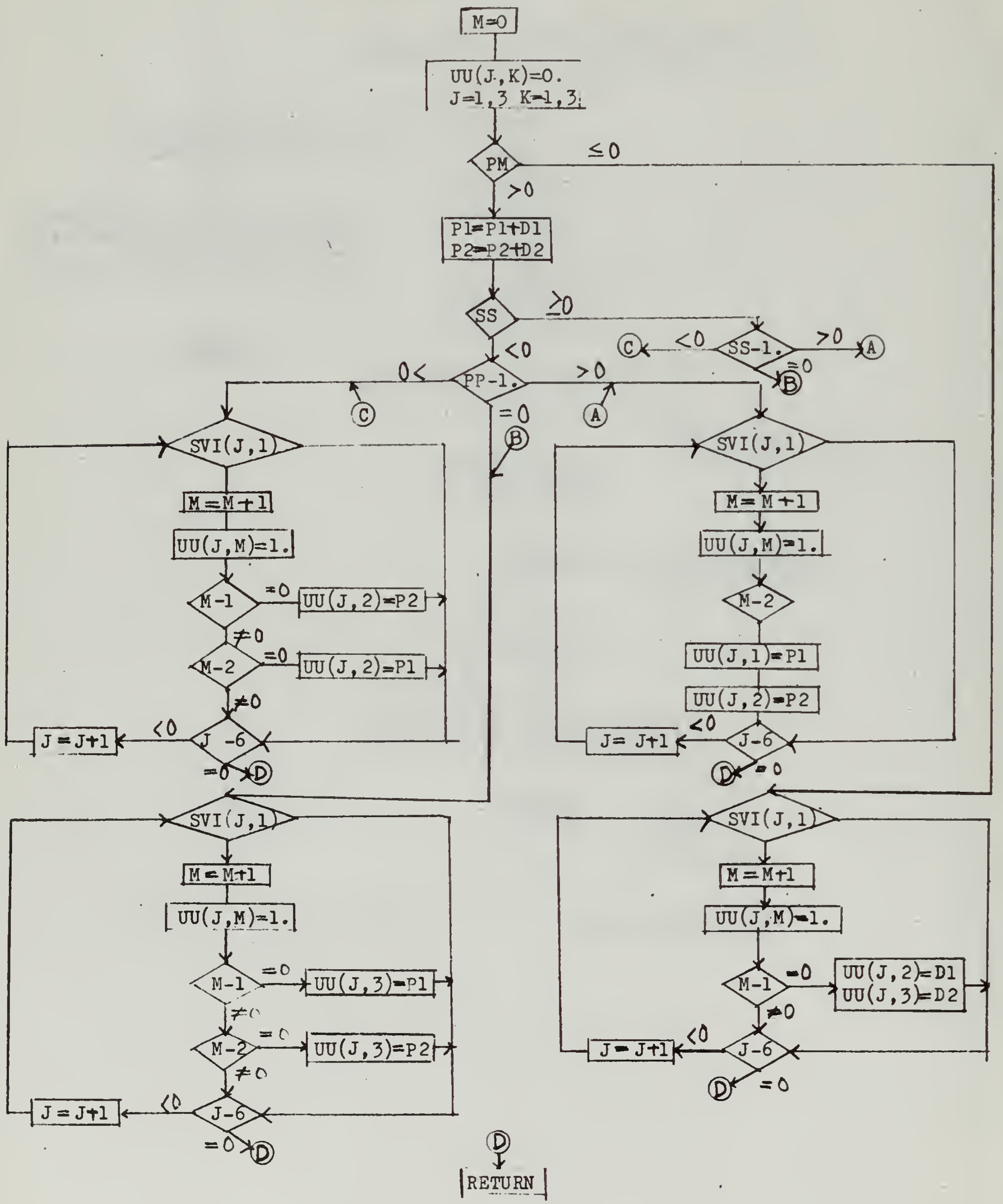
OUTPUT FLOW DIAGRAM(CONT.)



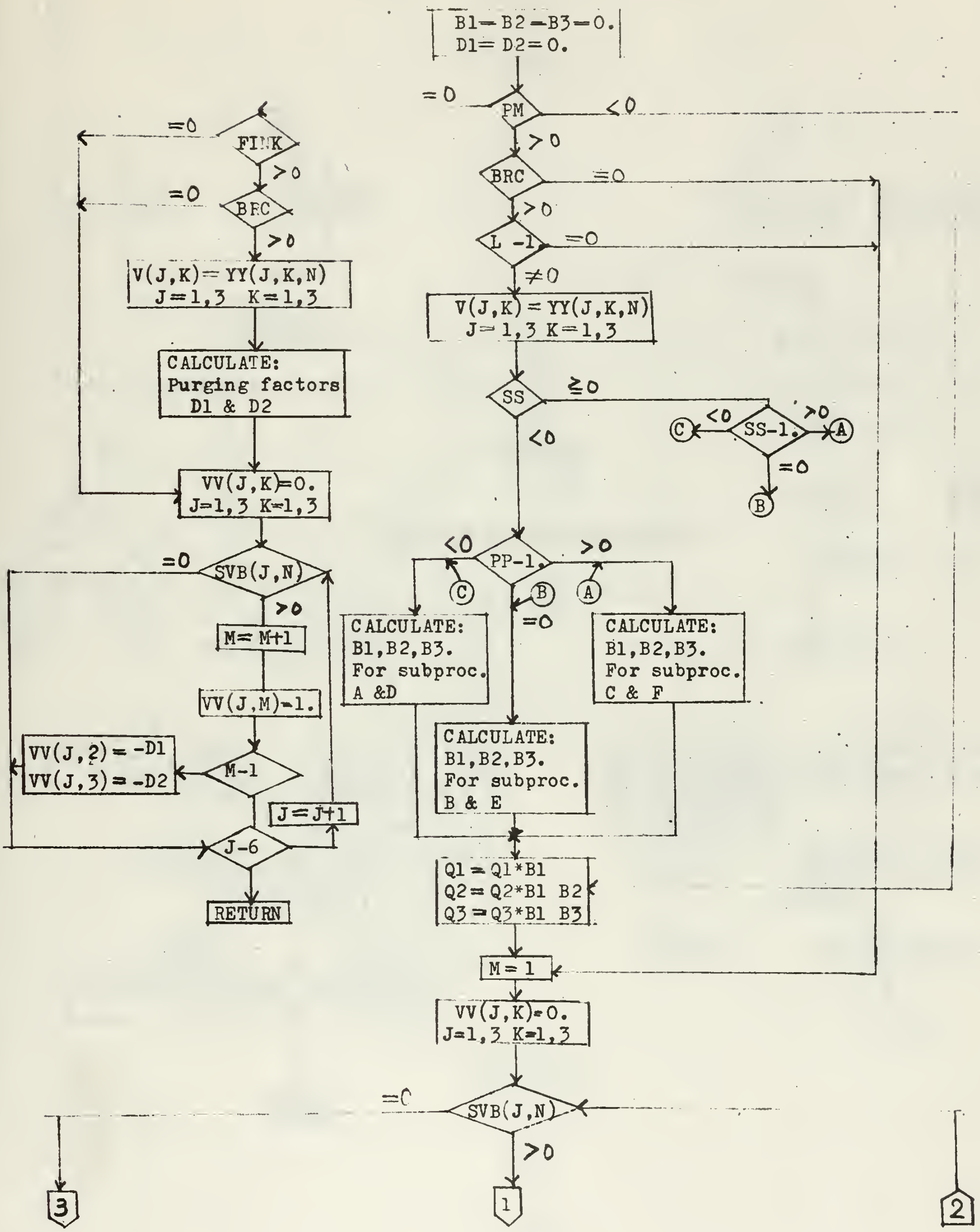
SUBROUTINE FINMAT FLOW DIAGRAM



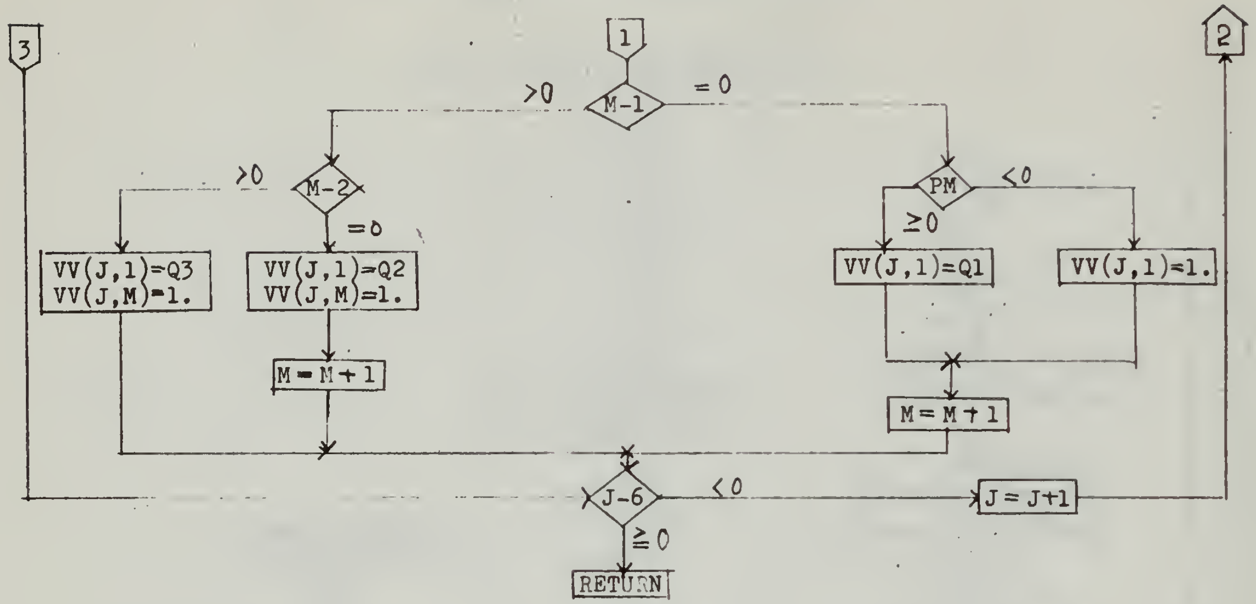
SUBROUTINE STATFM FLOW DIAGRAM



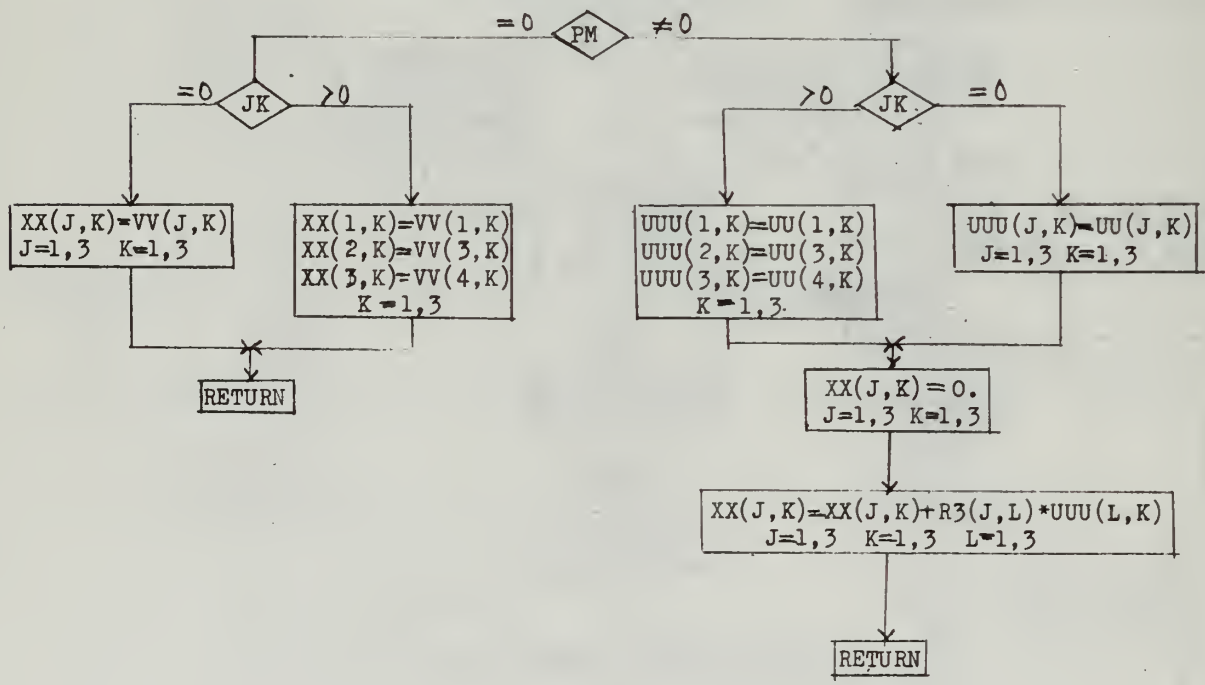
SUBROUTINE STATEB FLOW DIAGRAM



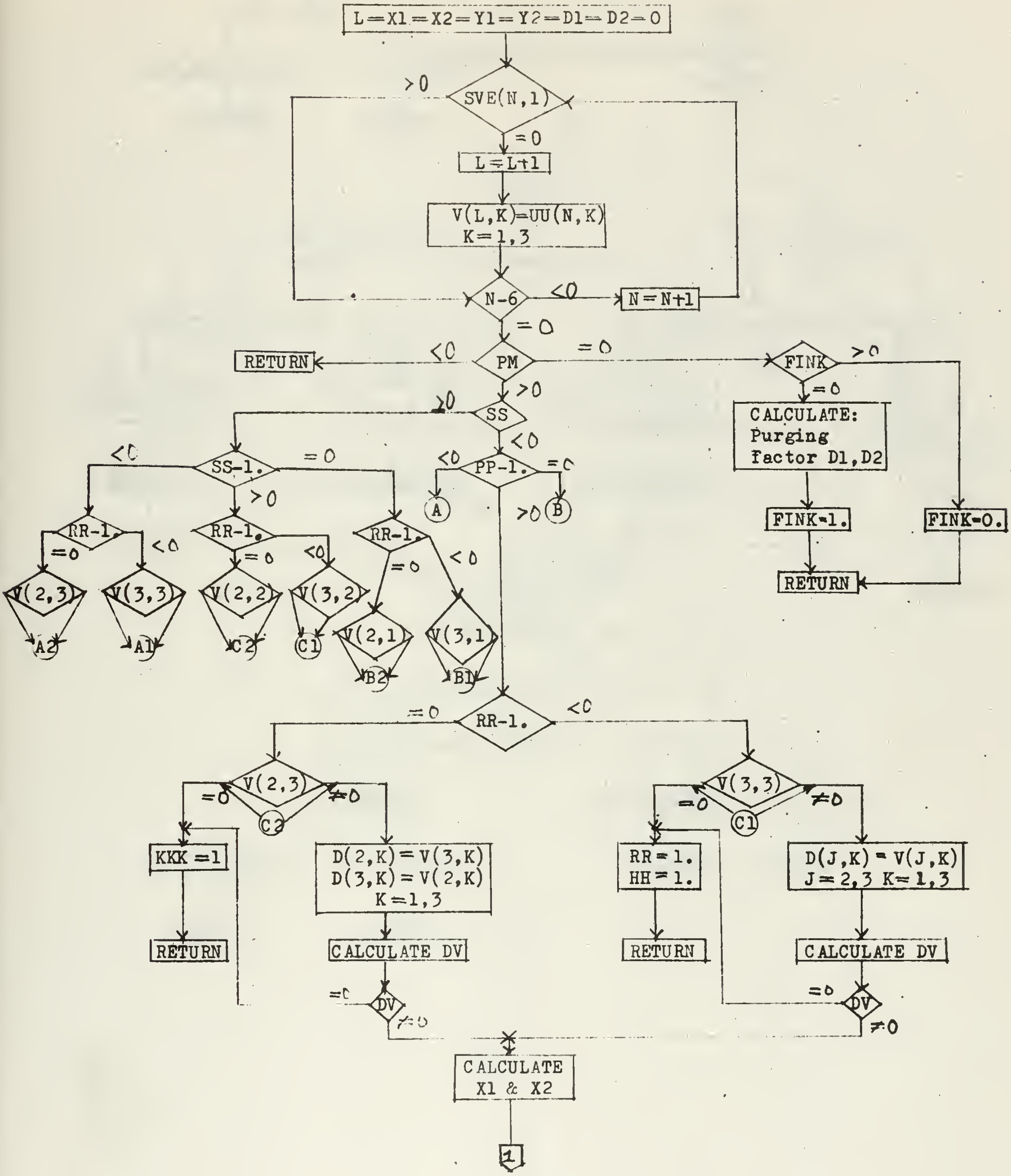
SUBROUTINE STATEB FLOW DIAGRAM(CONT.)

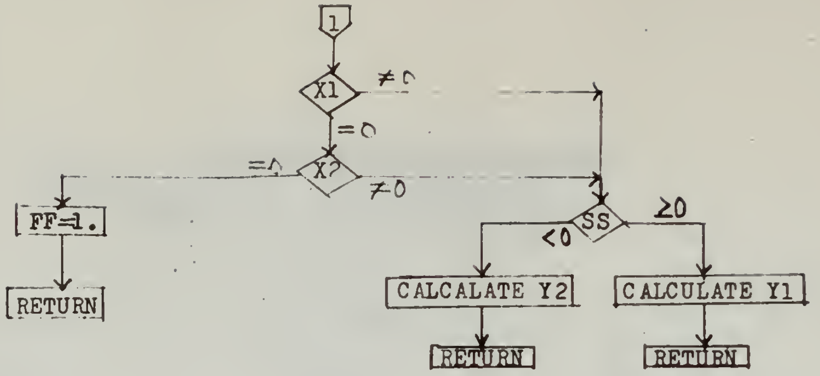


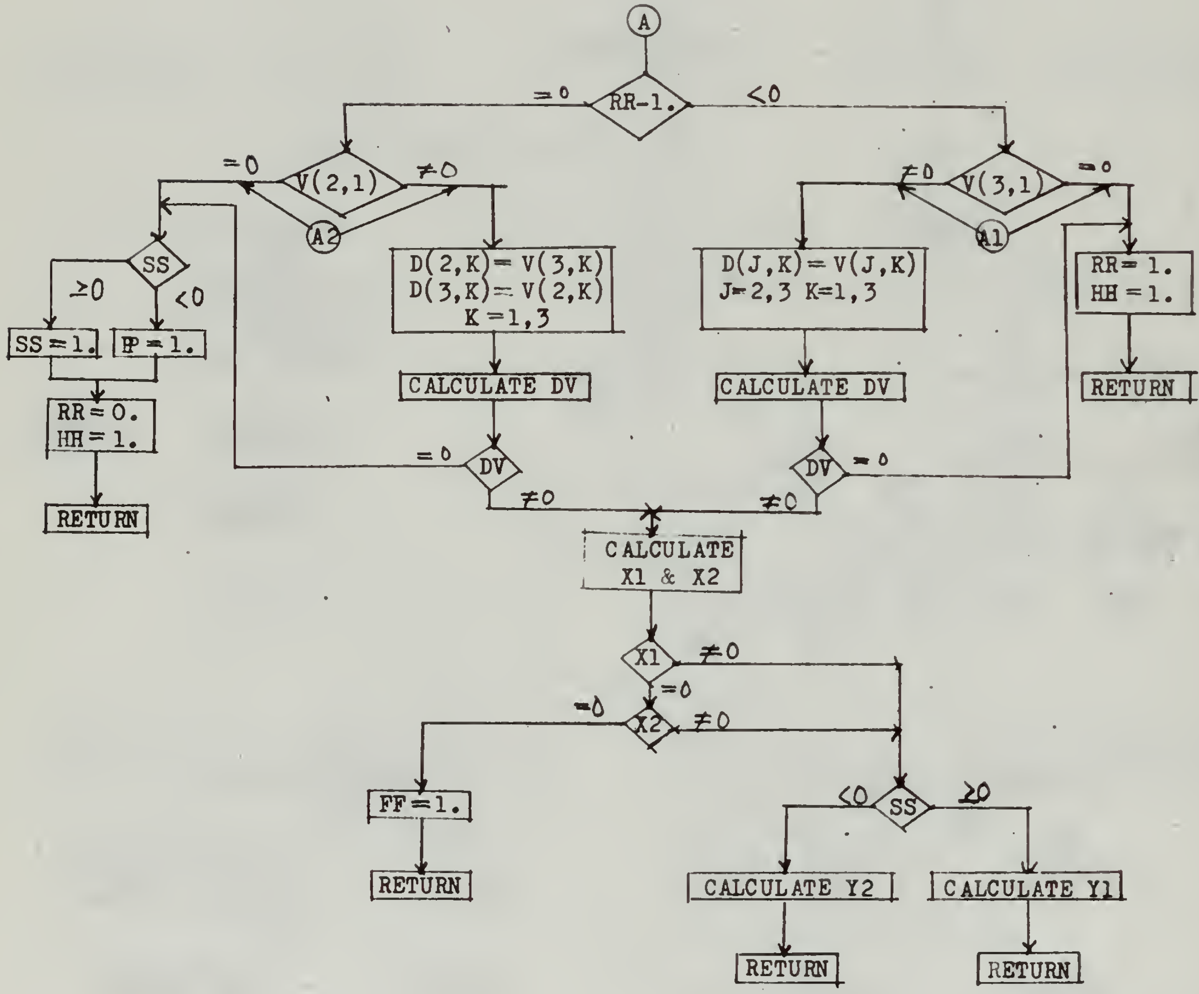
SUBROUTINE BRCOR FLOW DIAGRAM



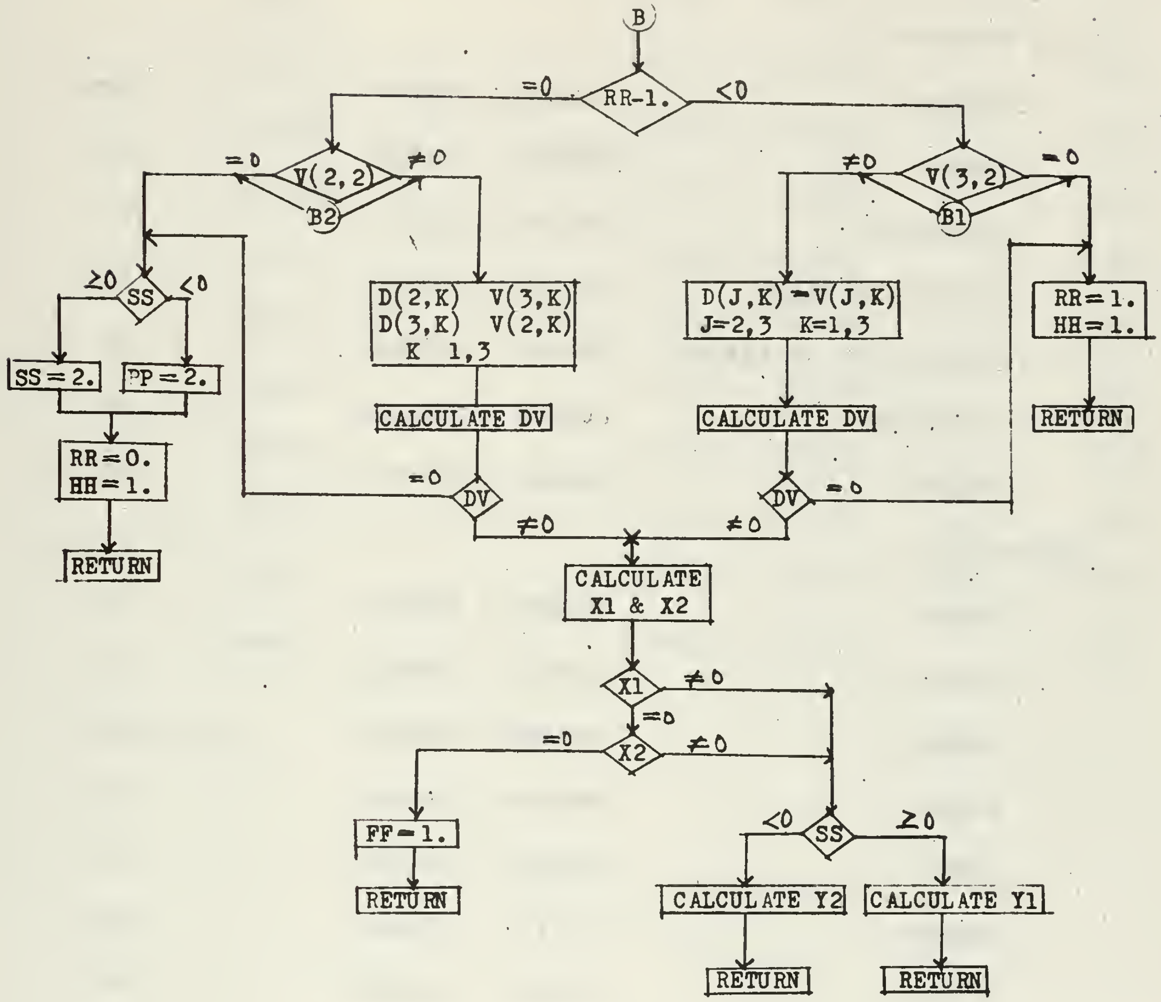
SUBROUTINE DELMA FLOW DIAGRAM




1
X1
≠0
=0
X2
=0
≠0
FF=1.
RETURN
SS
<0
≥0
CALCALATE Y2
CALCULATE Y1
RETURN
RETURN


A
RR-1.
=0
<0
V(2,1)
=0
≠0
A2
V(3,1)
≠0
=0
A1
SS
≥0
<0
SS=1.
PP=1.
RR=0.
HH=1.
RETURN
D(2,K)=V(3,K)
D(3,K)=V(2,K)
K=1,3
CALCULATE DV
DV
=0
≠0
D(J,K)=V(J,K)
J=2,3 K=1,3
CALCULATE DV
DV
=0
≠0
RR=1.
HH=1.
RETURN
CALCULATE
X1 & X2
X1
≠0
=0
X2
=0
≠0
FF=1.
RETURN
SS
<0
≥0
CALCULATE Y2
CALCULATE Y1
RETURN
RETURN

SUBROUTINE DELMA FLOW DIAGRAM(CONT.)



APPENDIX D

PROGRAM LISTING

D-1 INDEX

Main Program.

Section	Sequence Number	Page
INPUT	000000 - 001070	82
INVARIANTS	001080 - 001500	83
CONTROL IN PLANE	001510 - 002450	84
CONTROL OUT OF PLANE	002460 - 003390	86
ITERATION	003400 - 004420	88
OUTPUT	004430 - 007990	89

Subroutines

SUBSEC	008000 - 008520	96
DISTM	008530 - 009090	97
DISTMO	009100 - 009660	98
RIGID	009670 - 009920	99
RIGIO	009930 - 010180	99
STIFCO	010190 - 010430	100
STIFOO	010440 - 010680	100
STAVEC	010690 - 010990	101
STAVEO	011000 - 011260	101
INVERT	011270 - 012000	102
HANGER	012010 - 012180	103
HANGEO	012190 - 012360	103
CFIELO	012370 - 012880	104
POINO	012890 - 013100	105

Section	Sequence Number	Page
CFIELD	013110 - 013560	105
POINT	013570 - 013790	106
SFIELD	013800 - 014150	106
SFIELO	014160 - 014420	107
MATMUL	014430 - 014560	107
FINBRA	014570 - 014850	108
FINMAT	014860 - 015140	108
BRANCH	015150 - 015780	109
BRANCO	015790 - 016470	110
STATEM	016480 - 017020	111
STATEB	017030 - 017780	112
BRCOR	017790 - 018100	113
DELMA	018110 - 019500	114

D-2 Listing.

```
      PROGRAM VIPIPE
C
C
C     INPUT
C
C
      DIMENSION AJT(50),AJ(50),R(50),AIX(50),AIY(50),PHI(20),SVI(6,1),
     1DT(50),D(50),DI(50),TL(50),Z(50),G(50),E(50),AMU(50),THETA(50),RHO
     2(50),OMEGAM(20),DEL(600),OMEGA(600),A(6,6),B(6,6),U(6,6),CLX(50),C
     3LY(50),CLZ(50),CTX(50),CTY(50),CTZ(50),V(6,3),VV(6,3),UU(6,3),SVE(
     46,1),SV(22),SVIP(6,22),SVO(22),SVOP(6,22),NNBR(50),BAMU(50),THETR(
     520),P(20),YY(3,3,23),R3(3,3),XX(3,3),ITITLE(12),EDL(600)
      TYPE DOUBLE VV,UU,V,X2,X1,Y2,Y1,P1,P2,D1,D2,DEL
      NM=0
      JJ=1
  450 READ 1500,NS,NBR,IOP,HM,AMYK,SD,RI,OMEGAG,OMEGAI,GDABC,BRC,GRAPH,
     1REM,PMM,HL
 1500 FORMAT(3I2,4F3.0,2F8.0,4F2.0,2F4.0)
      IF (NS) 600,600,601
  601 NMBR=NBR+2
      IF(IOP-1)1501,1501,1506
 1501 READ 1502,(SV(I),I=1,NMBR)
 1502 FORMAT(22F3.0)
      DO 1505 J=1,NMBR
      IF(SV(J))1503,1503,1505
 1503 READ 1504,(SVIP(I,J),I=1,6)
 1504 FORMAT(6F2.0)
 1505 CONTINUE
      IF(IOP) 1506,1506,1523
 1506 READ 1507,(SVO(I),I=1,NMBR)
 1507 FORMAT(22F3.0)
      DO 1510 J=1,NMBR
      IF(SVO(J))1508,1508,1510
 1508 READ 1509,(SVOP(I,J),I=1,6)
 1509 FORMAT(6F2.0)
 1510 CONTINUE
 1523 IF(NBR) 1513,1513,1511
 1511 READ 1512,(PHI(I),I=1,NBR)
 1512 FORMAT (10F7.3)
 1513OREAD 1514,(D(I),DI(I),RHO(I),TL(I),THETA(I),CLX(I),CLY(I),CLZ(I),C
     1TX(I),CTY(I),CTZ(I),NNBR(I),I=1,NS)
 1514 FORMAT(3F5.2,2F6.2,6F7.2,I2)
      READ 1515 , AAMU,AE,AG,DD
 1515 FORMAT (F7.3,2E10.2,F2.0)
      IF(DD)1516,1516,1518
 1516 DO 1517 I=1,NS
      AMU(I)=AAMU
      E(I)=AE
 1517 G(I)=AG
      GO TO 1520
 1518 AMU(1)=AAMU
      E(1)=AE
      G(1)=AG
      READ 1519,(AMU(I),E(I),G(I),I=2,NS)
 1519 FORMAT (F7.3,2E10.2)
 1520 IF(OMEGAI) 1121,1121,1521
```

```
 1121 OMEGAI=.01
 1521 OMEGA(1)=OMEGAI
      AMYK=AMYK-1.0
      OMEGAO=OMEGAI
      FFAC=.5
      L=1
      M=1
      MN=0
      AA=0.0
      BB=0.0
      CC=0.0
      FF=0.0
      GG=0.0
      RR=0.0
      KK=0
      PP=0.
      KA=0
      KKK=0
      AAMU=0.0
      EE=0.0
      AL=0.0
      AD=0.0
      NNN=NBR+1
      IF(PMM*0.1-1.) 6091,6092,6093
 6091 PM=-1.
      GO TO 7000
 6092 PM=0.
      GO TO 7000
 6093 IF(PMM*0.01-1.) 6094,6095,6096
 6094 PM=-1.
      GO TO 7000
 6095 PM=1.
      GO TO 7000
 6096 IF(PMM*0.01-1.1) 6097,6098,6099
 6097 PM=-1.
      GO TO 7000
 6098 PM=0.
      GO TO 7000
 6099 PM=-1.
 7000 PPM=PM
      IF(PM) 7073,7073,7001
 7001 IF(REM-1.) 7071,7072,7071
 7071 SS=-1.
      GO TO 7073
 7072 SS=0.
 7073 DO 1522 I=1,NS
      DT(I)=DI(I)
      BAMU(I)=AMU(I)
      THETR(I)=THETA(I)
 1522 DI(I)=D(I)-2.0*DI(I)
C
C
C      INVARIANTS
C
      DO 108 I=1,NS
      IF(CLX(I)+CLY(I)+CLZ(I)+CTX(I)+CTY(I)+CTZ(I))108,101,108
```

```
  101 AMU(I)=AMU(I)*3.141592654*(D(I)**2-DI(I)**2)/(6912.*32.17*12.)
      IF(RHO(I))102,103,102
  102 THETA(I)=THETA(I)*3.141592654/180.
      TL(I)=THETA(I)*ABSF(RHO(I))
  103 AJT(I)=3.141592654*(D(I)**4-DI(I)**4)/32.
      AJ(I)=AJT(I)/2.
      R(I)=1.0/(AJ(I)*E(I))
      AIX(I)=SQRTF((D(I)**2+DI(I)**2)/8.)
      AIY(I)=AIX(I)/1.414214
      IF(AMYK)104,105,104
  104 Z(I)=1.0
      IF(RHO(I))105,106,105
  105 HHM=HM
      HMH=0.
      CALL SUBSEC (RHO(I),THETA(I),D(I),TL(I),HM,Z(I),OMEGAI,HMH)
      HM=HHM
  106 IF(NNBR(I))108,107,108
  107 AL=AL+TL(I)
      AD=AD+D(I)*TL(I)
      EE=EE+E(I)*TL(I)
      AID=AID+DI(I)*TL(I)
      AAMU=AAMU+AMU(I)*TL(I)
  108 CONTINUE
      AD=AD/AL
      AID=AID/AL
      AFR=3.141592654/64.*(AD**4-AID**4)
      FR=4.73004019**2*SQRTF(EE*AFR/(AAMU*AL**4))
      ADOM=FR/4.0
      DOM=ADOM
      CR=ADOM*0.001
      BDOM=ADOM
      BR=CR
      ERM=REM
      DO 109 II=1,NBR
  109 PHI(II)=PHI(II)*3.141592654/180.0
      IF(IOP-1) 110,999,998
  110 MN=1
C
C
C     CONTROL  IN PLANE
C
  999 P1=1.0
      P2=1.0
      D1=0.
      D2=0.
      Q1=1.0
      Q2=1.0
      Q3=1.0
      B1=0.
      B2=0.
      B3=0.
      FINK=0.
      IF(SV(1))1011,1021,1011
 1011 CALL STAVEC (SV(1),1,SVI)
      GO TO 1041
 1021 DO 1031 J=1,6
```

```
1031 SVI(J,1)=SVIP(J,1)
1041 IF(SV(NMBR))1051,1061,1051
1051 CALL STAVEC(SV(NMBR),1,SVE)
     GO TO 1081
1061 DO 1071 J=1,6
1071 SVE(J,1)=SVIP(J,NMBR)
1081 JK=0
     IF(NBR)1000,1000,1085
1085 DO 1087 N=2,NNN
     IF(SV(N))1087,1087,1086
1086 CALL STAVEC(SV(N),N,SVIP)
1087 CONTINUE
1000 N=2
     II=1
     HH=0.
     DO 1001 I=1,NS
     MAT=0
     BC=0.
     P(I)=CLX(I)+CLY(I)+CTZ(I)
     IF(P(I))37,38,37
  37 CALL HANGER(CLX(I),CLY(I),CTZ(I),U)
     GO TO 62
  38 IF(AMYK)39,41,39
  39 IF(RHO(I))40,42,40
  40 IF(Z(I))43,45,43
  41 IF(RHO(I))40,48,40
  42 IF(Z(I))46,47,46
  43 CALL CFIELD (R(I),Z(I),G(I),SD,RHO(I),THETA(I),D(I),DI(I),E(I),A)
  44 CALL POINT (AMU(I),AIY(I),OMEGA(L),TL(I),Z(I),RI,B)
     CALL MATMUL (A,B,U)
     MAT=1
     GO TO 62
  45 CALL STIFCO (THETA(I),RHO(I),U)
     GO TO 62
  460CALL DISTM (AMU(I),AIY(I),OMEGA(L),TL(I),R(I),D(I),DI(I),G(I),SD,R
    1I,E(I),FFAC,U)
     GO TO 62
  47 CALL RIGID (AMU(I),TL(I),OMEGA(L),AIY(I),RI,U)
     GO TO 62
  48 IF(Z(I))49,47,49
  49 CALL SFIELD (DI(I),TL(I),R(I),Z(I),G(I),SD,D(I),E(I),FFAC,A)
     GO TO 44
  62 IF(NNBR(I)) 81,80,81
  80 IF(I-1)63,79,63
  79 CALL STATEM(SVI,UU,D1,D2,P1,P2,PP,PM,SS)
     GO TO 63
  63 CALL FINMAT(U,UU,V,MAT,A,Z(I))
     DO 64 J=1,6
     DO 64 K=1,3
  64 UU(J,K)=V(J,K)
     IF(PM) 88,1113,84
1113 IF(FINK) 88,84,88
  84 IF(BC) 88,88,1111
1111 IF(BRC) 88,88,89
  89 CALL BRCOR(R3,UU,XX,JK,VV,PM)
     MMM=N-1
```

```
      DO 90 J=1,3
      DO 90 K=1,3
   90 YY(J,K,MMM)=XX(J,K)
   88 GO TO 1001
   81 IF(NNBR(I)-1) 83,83,65
   83 CALL STATEB(SVIP,VV,N,YY,L,BRC,D1,D2,PP,PM,SS,Q1,Q2,Q3,FINK)
   65 CALL FINBRA(U,VV,V,MAT,A,Z(I))
      DO 66 J=1,6
      DO 66 K=1,3
   66 VV(J,K)=V(J,K)
      IF(NNBR(I))68,67,67
   67 IF(NNBR(I)-3)1001,68,68
   68 CALL BRANCH(R3,VV,PHI(II),U)
      N=N+1
      II=II+1
      MAT=0
      BC=1.
      GO TO 63
 1001 CONTINUE
      GO TO 2000
C
C
C     CONTROL  OUT OF PLANE
C
  998 P1=1.0
      P2=1.0
      D1=0.
      D2=0.
      Q1=1.0
      Q2=1.0
      Q3=1.0
      B1=0.
      B2=0.
      B3=0.
      FINK=0.
      IF(SVO(1))1013,1023,1013
 1013 CALL STAVEO(SVO(1),1,SVI)
      GO TO 1043
 1023 DO 1033 J=1,6
 1033 SVI(J,1)=SVOP(J,1)
 1043 IF(SVO(NMBR))1053,1063,1053
 1053 CALL STAVEO(SVO(NMBR),1,SVE)
      GO TO 1083
 1063 DO 1073 J=1,6
 1073 SVE(J,1)=SVOP(J,NMBR)
 1083 JK=1
      IF(NBR)1002,1002,1093
 1093 DO 1095 N=2,NNN
      IF(SVO(N))1095,1095,1094
 1094 CALL STAVEO(SVO(N),N,SVOP)
 1095 CONTINUE
 1002 N=2
      II=1
      HH=0.
      DO 1003 I=1,NS
      BC=0.
```

```
      MAT=0
      P(I)=CTX(I)+CLZ(I)+CTY(I)
      IF(P(I))1,2,1
    1 CALL HANGEO(CTX(I),CLZ(I),CTY(I),U)
      GO TO 14
    2 IF(AMYK)3,11,3
    3 IF(RHO(I))4,8,4
    4 IF(Z(I))6,5,6
    5 CALL STIFOO (THETA(I),RHO(I),U)
      GO TO 14
    60CALL CFIELO(AJT(I),R(I),Z(I),G(I),SD,RHO(I),THETA(I),D(I),DI(I),FF
     1AC,A)
    7 CALL POINO (AMU(I),AIX(I),AIY(I),OMEGA(L),TL(I),Z(I),RI,B)
      CALL MATMUL (A,B,U)
      MAT=1
      GO TO 14
    8 IF(Z(I))10,9,10
    9 CALL RIGIO (AMU(I),TL(I),AIX(I),OMEGA(L),AIY(I),AJT(I),RI,U)
      GO TO 14
   100CALL DISTMO(AMU(I),AIX(I),AIY(I),OMEGA(L),TL(I),AJT(I),R(I),D(I),D
     1I(I),G(I),SD,RI,FFAC,U)
      GO TO 14
   11 IF(RHO(I))4,12,4
   12 IF(Z(I))13,9,13
   13 CALL SFIELO (DI(I),TL(I),AJT(I),R(I),Z(I),G(I),SD,D(I),FFAC,A)
      GO TO 7
   14 IF(NNBR(I)) 201,200,201
  200 IF(I-1) 15,202,15
  202 CALL STATEM(SVI,UU,D1,D2,P1,P2,PP,PM,SS)
      GO TO 15
   15 CALL FINMAT(U,UU,V,MAT,A,Z(I))
      DO 16 J=1,6
      DO 16 K=1,3
   16 UU(J,K)=V(J,K)
      IF(PM) 207,1114,85
 1114 IF(FINK) 207,85,207
   85 IF(BC)  207,207,1112
 1112 IF(BRC) 207,207,208
  208 CALL BRCOR(R3,UU,XX,JK,VV,PM)
      MMM=N-1
      DO 209 J=1,3
      DO 209 K=1,3
  209 YY(J,K,MMM)=XX(J,K)
  207 GO TO 1003
  201 IF(NNBR(I)-1)  204,204,17
  204 CALL STATEB(SVOP,VV,N,YY,L,BRC,D1,D2,PP,PM,SS,Q1,Q2,Q3,FINK)
   17 CALL FINBRA(U,VV,V,MAT,A,Z(I))
      DO 18 J=1,6
      DO 18 K=1,3
   18 VV(J,K)=V(J,K)
      IF(NNBR(I))82,19,19
   19 IF(NNBR(I)-3)1003,82,82
   82 CALL BRANCO(R3,VV,PHI(II),U)
      N=N+1
      II=II+1
      MAT=0
```

```
      BC=1.
      GO TO 15
 1003 CONTINUF
C
C        ITERATION
C
 2000 CALL DELMA(UU,SVE,V,X2,Y2,KKK,X1,FF,PP,HH,RR,SS,Y1,REM,FINK,PM,D1,
     1D2)
      IF(FINK) 87,87,86
   87 IF(PM) 99,99,98
   98 IF(KKK) 7030,7030,301
  301 GO TO 6001
 7030 IF(FF-1.0) 7051,7060,7051
 7060 KK=1
      M=M-1
      GO TO 3000
 7051 IF(HH-1.0) 300,6001,300
  300 IF(SS) 303,302,302
  303 DEL(L)=X2-Y2
      D1=X1
      D2=(X2+Y2)/2.0
      GO TO 61
  302 DEL(L)=X1-Y1
      D1=(X1+Y1)/2.0
      D2=X2
      GO TO 61
  990DEL(L)=V(1,1)*V(2,2)*V(3,3)-V(1,3)*V(2,2)*V(3,1)+V(1,2)*V(2,3)*V(3
     1,1)-V(1,2)*V(2,1)*V(3,3)+V(1,3)*V(3,2)*V(2,1)-V(1,1)*V(3,2)*V(2,3)
      GO TO 61
   21 PDEL=DEL(L)
      GO TO 25
   22 IF(AA)23,23,24
   23 PDEL=DEL(L-1)
      GO TO 25
   24 PDEL=DELC
   25 IF(PDEL*DEL(L))26,26,27
   26 IF(AA)28,28,57
   27 BB=0
      IF(AA)35,35,52
   28 AA=1.0
      BB=1.0
      DELC=DEL(L-1)
      OMEGAC=OMEGA(L-1)
      GO TO 59
   290OMEGAM(M)=OMEGAC+(OMEGA(L)-OMEGAC)*ABSF(DELC)/(ABSF(DELC)+
     1ABSF(DEL(L)))
      GO TO 32
   30 OMEGA(L+1)=OMEGA(L)-0.9*DOM
      DOM=0.5*DOM
   31 L=L+1
      BL=L
      IF(HL-BL) 7053,7053,86
 7053 KA=1
      GO TO 3000
   86 IF(JK)1000,1000,1002
   32 BM=M
```

```
      IF(HM-BM)3000,3000,33
   33 M=M+1
      IF(M-3) 340,34,34
   34 ADOM=(OMEGAM(M-1)-OMEGAM(M-2))*0.1
  340 AA=0.0
      BB=0
      CC=0
      GG=0.0
      DOM=ADOM
      DELC=0
      OMEGA(L+1)=OMEGAM(M-1)+2.*CR
      OMEGAO=OMEGA(L+1)
      GO TO 31
   35 IF(OMEGA(L)-OMEGAG)53,60,60
   50 OMEGA(L+1)=OMEGA(L)-(OMEGA(L)-OMEGAC)/2.0
      IF(OMEGA(L)-OMEGA(L+1)-CR)51,51,31
   51 CC=1.0
      GO TO 31
   52 DELC=DEL(L)
      OMEGAC=OMEGA(L)
   53 OMEGA(L+1)=OMEGA(L)+DOM
      GO TO 31
   54 IF(DELC*DEL(L))29,29,55
   550OMEGAM(M)=OMEGA(L)+(OMEGA(L-1)-OMEGA(L))*ABSF(DEL(L))/(ABSF(DEL(L)
     1)+ABSF(DEL(L-1)))
      GO TO 32
   56 IF(OMEGA(L)-OMEGAO)21,21,22
   57 IF(BB)58,58,50
   58 BB=1.0
   59 IF(OMEGA(L)-OMEGA(L-1)-CR)29,29,30
   60 M=M-1
      GO TO 3000
   61 IF(GG)70,70,72
   70 IF(DEL(L))77,71,77
   71 GG=1.0
      OMEGA(L+1)=OMEGA(L)-CR
      GO TO 31
   72 IF(GG-1.0)73,73,74
   73 GG=2.0
      HDEL=DEL(L)
      OMEGA(L+1)=OMEGA(L)+2.*CR
      GO TO 31
   74 TDEL=DEL(L)*HDEL
      IF(TDEL)75,76,76
   75 OMEGAM(M)=OMEGA(L-2)
      GO TO 33
   76 KK=1
      M=M-1
      GO TO 3000
   77 IF(CC)56,56,54
C
C     OUTPUT
C
 3000 IF(NM)400,400,403
  400 PRINT 401
  401 FORMAT (1H1//)
```

```
      PRINT 402
  402 FORMAT (15H PROGRAM VIPIPE,35X, 15HY. S. KIM NHA-3,38X, 13H  MARCH
     1  1966//,10X,108H               MODAL FREQUENCIES OF A PLANAR PIPING
     2 SYSTEM ARE DETERMINED BY AN ITERATIVE PROCEDURE USING THE /29H ME
     3THOD OF TRANSFER MATRICES.///119H *********************************
     4*******************************************************************
     5*******************      //)
  403 IF(NM)4058,4058,404
  404 PRINT 405
  405 FORMAT (1H1//)
 4058 NM=1
 4059 IF(JK)4060,4060,4065
 4060 PRINT 4070,JJ
 4070 FORMAT (50X,8HPROBLEM I2,//35X,47HIN PLANE MODE FREQUENCIES (RADIA
     1NS PER SECOND)   //45X,4HMODE,15X,10H FREQUENCY /)
      GO TO 4079
 4065 PRINT 4075,JJ
 4075 FORMAT(50X,8HPROBLEM I2,//33X,51HOUT OF PLANE MODE FREQUENCIES (RA
     1DIANS PER SECOND)      //45X,4HMODE,15X,10H FREQUENCY      /)
 4079 MH=M
      PRINT 408,(M,OMEGAM(M),M=1,MH)
  408 FORMAT (46X,I2,17X,F14.8 /)
      IF(KA) 5000,5000,5010
 5010 PRINT 5020
 5020 FORMAT(10X,53HPROBLEM TERMINATED DUE TO OMEGA STORAGE LIMITATION.
     1  /)
 5000 IF(KK)4080,4090,4080
 4080 PRINT 4085
 4085 FORMAT(10X,69HPROBLEM TERMINATED DUE TO SIGNIFICANT FIGURE LIMITAT
     1ION OF COMPUTER.     /)
 4090 IF(AMYK)412,409,412
  409 PRINT 411
  411 FORMAT (16X,53H              A LUMPED MASS APPROACH WAS EMPLOYED, A
     1ND )
      GO TO 414
  412 PRINT 413
  413 FORMAT (24X,45HA DISTRIBUTED MASS APPROACH WAS EMPLOYED AND      )
  414 IF(SD)415,415,420
  415 IF(RI)416,416,418
  416 PRINT 417
  417 FORMAT (2X,67HTHE EFFECTS OF SHEAR DEFLECTION AND ROTARY INERTIA W
     1ERE CONSIDERED. /)
      GO TO 4610
  418 PRINT 419
  419 FORMAT (3X,63HTHE EFFECTS OF SHEAR DEFLECTION WERE CONSIDERED WHIL
     1E THOSE OF      /3X,35HROTATIONAL INERTIA WERE NEGLECTED.        /)
      GO TO 4610
  420 IF(RI)421,421,423
  421 PRINT 422
  422 FORMAT (3X,72HTHE EFFECTS OF ROTATIONAL INERTIA WERE CONSIDERED WH
     1ILE THOSE OF SHEAR            /3X,27HDEFLECTION WERE NEGLECTED.   /)
      GO TO 4610
  423 PRINT 424
  424 FORMAT(3X,66HTHE EFFECTS OF SHEAR DEFLECTION AND ROTARY INERTIA WE
     1RE NEGLECTED. /)
 4610 IF(PM) 425,425,4713
```

```
4713 IF(SS) 4611,4612,4612
4611 IF(PP-1.) 4613,4614,4615
4613 PRINT 4623
4623 FORMAT(24X,35HPESTEL MAHRENHOLTZ METHOD(A) USED.    /)
     GO TO 425
4614 PRINT 4624
4624 FORMAT(24X,35HPESTEL MAHRENHOLTZ METHOD(B) USED.    /)
     GO TO 425
4615 PRINT 4625
4625 FORMAT(24X,35HPESTEL MAHRENHOLTZ METHOD(C) USED.    /)
     GO TO 425
4612 IF(SS-1.) 4616,4617,4618
4616 PRINT 4626
4626 FORMAT(24X,35HPESTEL MAHRENHOLTZ METHOD(D) USED.    /)
     GO TO 425
4617 PRINT 4627
4627 FORMAT(24X,35HPESTEL MAHRENHOLTZ METHOD(E) USED.    /)
     GO TO 425
4618 PRINT 4628
4628 FORMAT(24X,35HPESTEL MAHRENHOLTZ METHOD(F) USED.    /)
 425 IF(PM) 9010,9020,9000
9010 PRINT 9012
9012 FORMAT(24X,20HDELTA METHOD USED.   /)
     GO TO 9000
9020 PRINT 9022
9022 FORMAT(24X,29HMODIFIED DELTA METHOD USED.    /)
9000 PRINT 440
 4400FORMAT(118H *******************************************************
     1*************************************************************    )
     IF(GDABC-1.)4511,4520,4401
4401 PRINT 4402
44020FORMAT(35H SECTION LENGTH AND SUBSECTION DATA/23X,14HSECTION NUMBE
     1R,8X,25HLENGTH OF SECTION(INCHES),5X,21HNUMBER OF SECTIONS       )
     PRINT 443,(I,TL(I),Z(I),I=1,NS )
 443 FORMAT (29X,I3,23X,F7.2,21X,F4.0)
     LM=L
     IF(PM) 7054,7054,7055
7055 PRINT 441
 4410FORMAT (11H GRAPH DATA /22X,16HITERATION NUMBER,4X,30HFREQUENCY (R
     1ADIANS PER SECOND),3X,30H   VALUE OF REMAINDER                   )
     GO TO 7057
7054 PRINT 7056
70560FORMAT (11H GRAPH DATA /22X,16HITERATION NUMBER,4X,30HFREQUENCY (R
     1ADIANS PER SECOND),3X,30HVALUE OF FREQUENCY DETERMINANT          )
7057 PRINT 444,(L,OMEGA(L),DEL(L),L=1,LM)
 444 FORMAT (29X,I3,23X,F11.6,18X,E15.8)
4520 PRINT 442
 442 FORMAT(1X,10HINPUT DATA /)
     PRINT 4521
45210FORMAT(1X,9HCOMPONENT,4X,8HDIAMETER,5X,14HWALL THICKNESS ,5X,6HLEN
     1GTH,5X,9HRADIUS OF,9X,8HINCLUDED,5X,7HDENSITY,5X,7HELASTIC,5X,5HSH
     2EAR /57X,9HCURVATURE,5X,12HANGLE OF ARC,16X,7HMODULUS,5X,7HMODULUS
     3 /)
     DO 4525 I=1,NS
     IF(P(I))4524,4524,4523
4524 PRINT 4522,I,D(I),DT(I),TL(I),RHO(I),THETR(I),BAMU(I),E(I),G(I)
```

```
45220FORMAT (3X,I3,8X,F7.3,10X,F6.3,7X,F8.3,4X,F8.3,9X,F6.2,8X,F7.2,4X,
     1E8.2,4X,E8.2)
      GO TO 4525
 4523 SH=1.
 4525 CONTINUE
      IF(SH)4530,4530,4526
 4526 PRINT 4532
 4532 FORMAT (//1X,7HHANGERS /)
      PRINT 4527
45270FORMAT (2X,9HCOMPONENT,14X,3HCLX,14X,3HCLY,14X,3HCLZ,14X,3HCTX,14X
     1,3HCTY,14X,3HCTZ  /)
 4528 DO 4530 I=1,NS
      IF(P(I))4530,4530,4529
 4529 PRINT 4531,I,CLX(I),CLY(I),CLZ(I),CTX(I),CTY(I),CTZ(I)
 4531 FORMAT(3X,I3,15X,F10.2,7X,F10.2,7X,F10.2,7X,F10.2,7X,F10.2,7X,F10.
     12 )
 4530 CONTINUE
      PRINT 448
  448 FORMAT (//20H BOUNDARY CONDITIONS )
      PRINT 449,(SVI(I,1),I=1,6)
  449 FORMAT (5H SVI=6F3.0)
      PRINT 451,(SVE(I,1),I=1,6)
  451 FORMAT (5H SVE=6F3.0)
      IF(NBR)4511,4511,4502
 4502 IF(JK)4505,4505,4515
 4505 PRINT 4510,((SVIP(I,J),I=1,6),J=2,NNN)
 4510 FORMAT (6F3.0)
      GO TO 4511
 4515 PRINT 4517,((SVOP(I,J),I=1,6),J=2,NNN)
 4517 FORMAT (6F3.0)
 4511 CONTINUE
C     SORT
      IF(GRAPH-1.) 6001,6000,6001
 6000 LM1=LM-1
      DO 6100 L=1,LM
      EDL(L)=DEL(L)
 6100 CONTINUE
      DO 6600 L=1,LM1
      LP1=L+1
      DO 6600 M=LP1,LM
      IF(OMEGA(L)-OMEGA(M)) 6600,6600,6003
 6003 TEMP=OMEGA(L)
      OMEGA(L)=OMEGA(M)
      OMEGA(M)=TEMP
      TEMP=EDL(L)
      EDL(L)=EDL(M)
      EDL(M)=TEMP
 6600 CONTINUE
C     GRAPH
      NUMPTS=60
      K=0
      DO 6006 J=1,12
 6006 ITITLE(J)=8H
      ITITLE(3)=8H KIM Y S
      ITITLE(8)=8HFREQUENC
      ITITLE(9)=8HY  VS  R
```

```
      ITITLE(10)=8HEMAINDER
      LT1=LM/NUMPTS
      IF(LT1) 6024,6025,6024
 6025 NUMPTS=LM
      LABEL=4H
      L=1
      CALL DRAW(NUMPTS,OMEGA,EDL,0,0,LABEL,ITITLE,0,0,0,0,0,0,9,
     110,K,LAST)
      GO TO 6001
 6024 DO 6020 J=1,17
      LT2=0
      LT2=LT2+J*NUMPTS
      IF(LT2-LM) 6020,6021,6022
 6020 CONTINUE
 6021 LT=LT1
      GO TO 6023
 6022 LT=LT1+1
      NUMEND=LM-LT1*NUMPTS
 6023 L=1
      DO 7021 I=1,LT
      IF(I-1) 6008,6041,6008
 6008 IF(I-LT) 8011,6011,6011
 8011 IF(I-3) 6042,6043,8012
 8012 IF(I-5) 6044,6045,8013
 8013 IF(I-7) 6046,6047,8014
 8014 IF(I-9) 6048,6049,8015
 8015 IF(I-LT) 6050,6011,6001
 6041 LABEL=4H ONE
      GO TO 6009
 6042 LABEL=4H TWO
      GO TO 6010
 6043 LABEL=4HTHRE
      GO TO 6010
 6044 LABEL=4HFOUR
      GO TO 6010
 6045 LABEL=4HFIVE
      GO TO 6010
 6046 LABEL=4H SIX
      GO TO 6010
 6047 LABEL=4HSEVN
      GO TO 6010
 6048 LABEL=4H EIT
      GO TO 6010
 6049 LABEL=4HNINE
      GO TO 6010
 6050 LABEL=4H TEN
      GO TO 6010
 6009 DO 6031 N=1,NUMPTS
      EDL(N)=EDL(L+N-1)
 6031 OMEGA(N)=OMEGA(L+N-1)
      PRINT 6027,NUMPTS,EDL(1)
 6027 FORMAT(1H1,///////5X,7HNUMPTS=,I3,5X,7HEDL(1)=,E12.9)
      CALL DRAW(NUMPTS,OMEGA,EDL,0,0,LABEL,ITITLE,0,0,0,0,0,0,9,
     110,K,LAST)
      GO TO 7021
 6010 L=L+NUMPTS
```

```
      OMEGAX=OMEGA(L)
      DO 6030 N=1,NUMPTS
      EDL(N)=EDL(N+L-1)
      OMEGA(N)=OMEGA(L+N-1)
 6030 OMEGA(N)=OMEGA(N)-OMEGAX
      PRINT 6032,NUMPTS,EDL(1)
 6032 FORMAT(///////5X,7HNUMPTS=, I3   ,5X,7HEDL(1)=,E12.9)
      CALL DRAW(NUMPTS,OMEGA,EDL,0,0,LABEL,ITITLE,0,0,0,0,0,0,9,
     110,K,LAST)
      GO TO 7021
 6011 L=L+NUMPTS
      OMEGAX=OMEGA(L)
      DO 6033 N=1,NUMEND
      EDL(N)=EDL(N+L-1)
      OMEGA(N)=OMEGA(N+L-1)
 6033 OMEGA(N)=OMEGA(N)-OMEGAX
      PRINT 6051,NUMEND,EDL(1)
 6051 FORMAT(///////5X,7HNUMEND=,I3,    5X,7HEDL(1)=,E12.9)
      LABEL=4HLAST
      CALL DRAW(NUMEND,OMEGA,EDL,0,0,LABEL,ITITLE,0,0,0,0,0,0,9,
     110,K,LAST)
      GO TO 6001
 7021 CONTINUE
 6001 IF(HH-1.0) 8024,8023,8023
 8023 MH=M
      LM=L
 8024 DO 446 M=1,MH
  446 OMEGAM(M)=0.0
      DO 447 L=1,LM
      OMEGA(L)=0.0
      EDL(L)=0.
  447 DEL(L)=0.0
      DELC=0.0
      OMEGAC=0.0
      IF(HH-1.0) 8001,8022,8001
 8001 IF(PM) 8031,8031,8033
 8033 IF(REM) 8031,8032,8032
 8031 IF(MN)8022,8022,452
 8032 IF(REM-1.) 8041,8042,8043
 8041 IF(PP-1) 8043,8043,8072
 8072 IF(SS) 8073,8074,8074
 8073 SS=0.
      GO TO 8022
 8074 IF(SS-1.)8042,8042,8075
 8075 SS=-1.
      PP=0.
      REM=-1.
      GO TO 8033
 8042 SS=SS-1.
      IF(SS) 8051,8052,8053
 8051 SS=1.
      GO TO 8022
 8052 SS=2.
      GO TO 8022
 8053 SS=-1.
      REM=-1.
```

```
      GO TO 8033
 8043 PP=PP-1.
      IF(PP) 8061,8062,8063
 8061 PP=1.
      GO TO 8022
 8062 PP=2.
      GO TO 8022
 8063 PP=0.
      REM=-1.
      GO TO 8033
  452 PP=0.
      RR=0.
      SS=-1.
 8022 L=1
      M=1
      AA=0.0
      BB=0.0
      CC=0.0
      FF=0.0
      GG=0.0
      KK=0
      KA=0
      KKK=0
      ADOM=BDOM
      CR=BR
      OMEGA(1)=OMEGAI
      OMEGAO=OMEGAI
      DOM=ADOM
      IF(HH-1.0) 8021,7040,8021
 7040 IF(SS) 8191,8192,8192
 8191 IF(PP-1.) 8091,8092,8093
 8192 IF(SS-1.) 8291,8292,8293
 8091 PRINT 8081
 8081 FORMAT(1H1,////18HMETHOD  A  FAILED.   ///)
      GO TO 8021
 8092 PRINT 8082
 8082 FORMAT(1H1,////18HMETHOD  B  FAILED.   ///)
      GO TO 8021
 8093 PRINT 8084
 8084 FORMAT(1H1,////18HMETHOD  C  FAILED.   ///)
      GO TO 8021
 8291 PRINT 8281
 8281 FORMAT(1H1,////18HMETHOD  D  FAILED.   ///)
      GO TO 8021
 8292 PRINT 8282
 8282 FORMAT(1H1,////18HMETHOD  E  FAILED.   ///)
      GO TO 8021
 8293 PRINT 8283
 8283 FORMAT(1H1,////18HMETHOD  F  FAILED.   ///)
 8021 IF(PM) 453,453,8035
 8035 IF(REM) 8037,8036,8036
 8036 IF(JK) 998,999,998
 8037 REM=ERM
  453 IF(PM) 6081,6082,6080
 6081 IF(PMM-1.) 6084,6080,6084
 6084 IF(PMM*0.1-10.1) 6086,6085,6086
```

```
6085 PM=1.
     GO TO 8036
6086 PM=0.
     GO TO 8036
6082 IF(PMM*0.1-1.1) 6080,6080,6085
6080 IF(MN) 9001,9001,8034
8034 MN=0
     PM=PPM
     GO TO 998
9001 JJ=JJ+1
     DO 454 I=1,NS
     P(I)=0.
 454 Z(I)=0.0
     GO TO 450
 600 END
C
C
      SUBROUTINE SUBSEC (RHO,THETA,D,TL,HM,Z,OMEGAI,HMH)
      RHOD=ABSF(RHO)
      IF(HMH) 41,40,41
   40 IF(HM-5.0)8,8,7
    8 IF(OMEGAI-800.) 10,7,7
    7 HM=6.
      GO TO 10
   41 HM=HMH
   10 RA=TL/D
      IF(RHOD-0.)3,3,4
    3 IF(RA-1.)5,5,6
    5 Z=0.
      RETURN
    6 IF(RA-3.) 15,15,16
   15 Z=2.0
      RETURN
   16 IF(RA-6.)17,17,18
   17 Z=3.0
      RETURN
   18 IF(HM-3.)1,1,2
    1 Z=6.0
      RETURN
    2 Z=2.0*HM
      RETURN
    4 RD=RHOD/D
      IF(RD-1.)19,19,23
   19 Z=0
      RETURN
   23 IF(RD-3.)24,24,25
   24 IF(THETA-.44)26,26,27
   26 Z=0.
      RETURN
   27 IF(THETA-1.57)28,28,29
   28 Z=4.0
      RETURN
   29 Z=6.0
      RETURN
   25 IF(RD-6.)30,30,31
   30 IF(THETA-.44)32,32,33
```

```
32 Z=4.0
   RETURN
33 IF(THETA-1.57)34,34,35
34 Z=6.0
   RETURN
35 Z=8.0
   RETURN
31 IF(THETA-.44)36,36,3
36 Z=4.0
   RETURN
   END
C
C
   SUBROUTINE DISTM (AMU,AIY,OMEGA,TL,R,D,DI,G,SD,RI,E,FFAC,U)
   DIMENSION U(6,6)
   ARA=(3.141592654/4.0)*(D**2-DI**2)
   AREA=ARA*FFAC
   BE=TL*OMEGA*SQRTF(AMU/(E*ARA))
   P4=R*AMU*TL**4*OMEGA**2
   IF(RI)1,1,2
 1 T=R*AMU*(AIY*OMEGA*TL)**2
   GO TO 3
 2 T=0.0
 3 IF(SD)4,4,5
 4 S=(AMU*(OMEGA*TL)**2)/(G*AREA)
   GO TO 6
 5 S=0.0
 6 SPT=S+T
   VV=SQRTF(P4+(S-T)**2/4.0)
   AL1=SQRTF(VV-.5*SPT)
   AL2=SQRTF(VV+.5*SPT)
   CL=1.0/(AL1**2+AL2**2)
   CHL1=(EXPF(AL1)+1./EXPF(AL1))/2.
   SHL1=(EXPF(AL1)-1./EXPF(AL1))/2.
   CL2=COSF(AL2)
   SL2=SINF(AL2)
   SBE=SINF(BE)
   CBE=COSF(BE)
   C00=CL*(AL2**2*CHL1+AL1**2*CL2)
   C01=CL*((AL2**2*SHL1)/AL1+(AL1**2*SL2)/AL2)
   C02=CL*(CHL1-CL2)
   C03=CL*(SHL1/AL1-SL2/AL2)
   DO 7 J=1,6
   DO 7 K=1,6
 7 U(J,K)=0.0
   U(1,1)=CBE
   U(1,6)=TL*SBE/(BE*ARA*E)
   U(2,2)=C00-S*C02
   U(2,3)=TL*(C01-SPT*C03)
   U(2,4)=R*C02*TL**2
   U(2,5)=(R*TL**3/P4)*((P4+S**2)*C03-S*C01)
   U(3,2)=P4*C03/TL
   U(3,3)=C00-T*C02
   U(3,4)=TL*R*(C01-T*C03)
   U(3,5)=U(2,4)
   RR=1.0/R
```

```
      U(4,2)=P4*RR*CO2/TL**2
      U(4,3)=RR*((P4+T**2)*CO3-T*CO1)/TL
      U(4,4)=U(3,3)
      U(4,5)=U(2,3)
      U(5,2)=P4*RR*(CO1-S*CO3)/TL**3
      U(5,3)=U(4,2)
      U(5,4)=U(3,2)
      U(5,5)=U(2,2)
      U(6,1)=-AMU*TL*OMEGA**2*SBE/BE
      U(6,6)=CBE
      RETURN
      END
C
C
      SUBROUTINE DISTMO (AMU,AIX,AIY,OMEGA,TL,AJT,R,D,DI,G,SD,RI,FFAC,U)
      DIMENSION U(6,6)
      W=1./(G*AJT)
      AREA=(3.141592654/4.0)*(D**2-DI**2)*FFAC
      BE=+SQRTF(AMU * ( T L  *AIX*OMEGA)**2*W)
      P4=R*AMU   * TL **4*OMEGA**2
      IF(RI)2,2,3
    2 T=R*AMU*(AIY*OMEGA*TL)**2
      GO TO 4
    3 T=0.0
    4 IF(SD)5,5,6
    5 S=(AMU*(OMEGA*TL)**2)/(G*AREA)
      GO TO 7
    6 S=0.0
    7 VV=+SQRTF(P4+(S-T)**2/4.0)
      SPT=S+T
      AL1=+SQRTF(VV-.5*SPT)
      AL2=+SQRTF(VV+.5*SPT)
      CL=1.0/(AL1**2+AL2**2)
      CHL1=(EXPF(AL1)+1./EXPF(AL1))/2.
      SHL1=(EXPF(AL1)-1./EXPF(AL1))/2.
      CL2=COSF(AL2)
      SL2=SINF(AL2)
      SBE=SINF(BE)
      CBE=COSF(BE)
      CO0=CL*(AL2**2*CHL1 + AL1**2*CL2)
      CO1=CL*((AL2**2*SHL1)/AL1 + (AL1**2*SL2)/AL2)
      CO2=CL*(CHL1-CL2)
      CO3=CL*(SHL1/AL1-SL2/AL2)
      DO 1 J=1,6
      DO 1 K=1,6
    1 U(J,K)=0.0
      U(1,1)=CBE
      U(1,2)=TL*W*SBE/BE
      U(2,1)=-AMU  * TL  *AIX**2*SBE/BE
      U(2,2)=CBE
      U(3,3)=CO0-S*CO2
      U(3,4)= TL *(CO1-SPT*CO3)
      U(3,5)=R*CO2*TL**2
      U(3,6)=(R*TL**3/P4)*(-S*CO1+(P4+S**2)*CO3)
      U(4,3)=P4*CO3/TL
      U(4,4)=CO0-T*CO2
```

```
      U(4,5)= TL*R*(CO1-T*CO3)
      U(4,6)=U(3,5)
      RR=1.0/R
      U(5,3)=P4*RR*CO2/TL**2
      U(5,4)=RR*(-T*CO1+(P4+T**2)*CO3)/TL
      U(5,5)=U(4,4)
      U(5,6)=U(3,4)
      U(6,3)=P4*RR*(CO1-S*CO3)/TL**3
      U(6,4)=U(5,3)
      U(6,5)=U(4,3)
      U(6,6)=U(3,3)
      RETURN
      END
C
C
      SUBROUTINE RIGID (AMU,TL,OMEGA,AIY,RI,U)
      DIMENSION U(6,6)
      AM=AMU*TL
      DO 1 J=1,6
      DO 1 K=1,6
    1 U(J,K)=0.0
      U(1,1)=1.0
      U(6,1)=-AM*OMEGA**2
      U(6,6)=1.0
      U(2,2)=1.0
      U(2,3)=TL
      U(3,3)=1.0
      U(4,2)=AM*TL*OMEGA**2/2.0
      IF(RI)2,2,3
    2 U(4,3)=AM*OMEGA**2*(TL**2/6.0-AIY**2)
      GO TO 4
    3 U(4,3)=AM*(OMEGA*TL)**2/6.0
    4 U(4,4)=1.0
      U(4,5)=TL
      U(5,2)=-U(6,1)
      U(5,3)=U(4,2)
      U(5,5)=1.0
      RETURN
      END
C
C
      SUBROUTINE RIGIO (AMU,TL,AIX,OMEGA,AIY,AJT,SDRI,U)
      DIMENSION U(6,6)
      DO 2 J=1,6
      DO 2 K=1,6
    2 U(J,K)=0.0
      AM=AMU*TL
      U(1,1)=1.0
      U(2,1)=-AMU*TL*(AIX*OMEGA)**2
      U(2,2)=1.0
      U(3,3)=1.0
      U(3,4)=TL
      U(4,4)=1.0
      U(5,3)=AM*TL*OMEGA**2/2.0
      IF(SDRI-0.) 30,30,31
   30 U(5,4)=AM*OMEGA**2*(TL**2/6.0-AIY**2)
```

```
      GO TO 1
   31 U(5,4)=AM*OMEGA**2*(TL**2/6.0)
    1 U(5,5)=1.0
      U(5,6)=TL
      U(6,3)=AM*OMEGA**2
      U(6,4)=U(5,3)
      U(6,6)=1.0
      RETURN
      END
C
C
      SUBROUTINE STIFCO (THETA,RHO,U)
      DIMENSION U(6,6)
      DO 1 J=1,6
      DO 1  K=1,6
    1 U(J,K)=0.0
      IF(RHO)2,2,3
    2 V=-1.0
      GO TO 4
    3 V=1.0
    4 CT=COSF(THETA)
      ST=SINF(THETA)
      U(1,1)=CT
      U(1,2)=-V*ST
      U(2,1)=V*ST
      U(2,2)=CT
      U(3,3)=1.0
      U(4,4)=1.0
      U(5,5)=CT
      U(5,6)=-V*ST
      U(6,5)=V*ST
      U(6,6)=CT
      RETURN
      END
C
C
      SUBROUTINE STIFOO (THETA,RHO,U)
      DIMENSION U(6,6)
      IF(RHO)1,1,2
    1 V=-1.0
      GO TO 3
    2 V=1.0
    3 CT=COSF(THETA)
      ST=SINF(THETA)
      DO 4 J=1,6
      DO 4 K=1,6
    4 U(J,K)=0.0
      U(1,1)=CT
      U(1,4)=ST*V
      U(2,2)=CT
      U(2,5)=ST*V
      U(3,3)=1.0
      U(4,1)=-ST*V
      U(4,4)=CT
      U(5,2)=-ST*V
      U(5,5)=CT
```

```
      U(6,6)=1.0
      RETURN
      END
C
C
      SUBROUTINE STAVEC(SV,N,BC)
      DIMENSION BC(6,22)
      DO 10 J=1,6
   10 BC(J,N)=0.
      IF(SV-1.)1,1,2
    1 BC(4,N)=1.0
      BC(5,N)=1.0
      BC(6,N)=1.0
      RETURN
    2 IF(SV-2.)3,3,4
    3 BC(1,N)=1.0
      BC(2,N)=1.0
      BC(3,N)=1.0
      RETURN
    4 IF(SV-3.)5,5,6
    5 BC(3,N)=1.0
      BC(5,N)=1.0
      BC(6,N)=1.0
      RETURN
    6 IF(SV-4.)7,7,8
    7 BC(1,N)=1.0
      BC(3,N)=1.0
      BC(5,N)=1.0
      RETURN
    8 BC(2,N)=1.0
      BC(3,N)=1.0
      BC(6,N)=1.0
      RETURN
      END
C
C
      SUBROUTINE STAVEO (SV,N,BC)
      DIMENSION BC(6,22)
      DO 10 L=1,6
   10 BC(L,N)=0.0
      IF(SV-1.0)1,1,2
    1 BC(2,N)=1.0
      BC(5,N)=1.0
      BC(6,N)=1.0
      RETURN
    2 IF(SV-2.0)3,3,4
    3 BC(1,N)=1.0
      BC(3,N)=1.0
      BC(4,N)=1.0
      RETURN
    4 IF(SV-3.0)5,5,6
    5 BC(2,N)=1.0
      BC(4,N)=1.0
      BC(6,N)=1.0
      RETURN
    6 IF(SV-4.0)7,7,5
```

```
    7 BC(1,N)=1.0
      BC(4,N)=1.0
      BC(6,N)=1.0
      RETURN
      END
C
C
      SUBROUTINE INVERT (A,N,D,L,M)
C     PROGRAM FOR FINDING THE INVERSE OF A NXN MATRIX
C
      DIMENSION A(25,25),L(25),M(25)
C     SEARCH FOR LARGEST ELEMENT
      D=1.0
      DO80 K=1,N
      L(K)=K
      M(K)=K
      BIGA=A(K,K)
      DO20 I=K,N
      DO20 J=K,N
      IF(ABSF(BIGA)-ABSF(A(I,J))) 10,20,20
   10 BIGA=A(I,J)
      L(K)=I
      M(K)=J
   20 CONTINUE
C     INTERCHANGE ROWS
      J=L(K)
      IF(L(K)-K) 35,35,25
   25 DO30 I=1,N
      HOLD=-A(K,I)
      A(K,I)=A(J,I)
   30 A(J,I)=HOLD
C     INTERCHANGE COLUMNS
   35 I=M(K)
      IF(M(K)-K) 45,45,37
   37 DO40 J=1,N
      HOLD=-A(J,K)
      A(J,K)=A(J,I)
   40 A(J,I)=HOLD
C     DIVIDE COLUMN BY MINUS PIVOT
   45 DO55 I=1,N
   46 IF(I-K)50,55,50
   50 A(I,K)=A(I,K)/(-A(K,K))
   55 CONTINUE
C     REDUCE MATRIX
      DO65 I=1,N
      DO65 J=1,N
   56 IF(I-K) 57,65,57
   57 IF(J-K) 60,65,60
   60 A(I,J)=A(I,K)*A(K,J)+A(I,J)
   65 CONTINUE
C     DIVIDE ROW BY PIVOT
      DO75 J=1,N
   68 IF(J-K)70,75,70
   70 A(K,J)=A(K,J)/A(K,K)
   75 CONTINUE
C     CONTINUED PRODUCT OF PIVOTS
```

```
      D=D*A(K,K)
C     REPLACE PIVOT BY RECIPROCAL
      A(K,K)=1.0/A(K,K)
   80 CONTINUE
C     FINAL ROW AND COLUMN INTERCHANGE
      K=N
  100 K=(K-1)
      IF(K) 150,150,103
  103 I=L(K)
      IF(I-K) 120,120,105
  105 DO110 J=1,N
      HOLD=A(J,K)
      A(J,K)=-A(J,I)
  110 A(J,I)=HOLD
  120 J=M(K)
      IF(J-K) 100,100,125
  125 DO130 I=1,N
      HOLD=A(K,I)
      A(K,I)=-A(J,I)
  130 A(J,I)=HOLD
      GO TO 100
  150 RETURN
      END
C
C
      SUBROUTINE HANGER (CLX,CLY,CTZ,U)
      DIMENSION U(6,6)
      DO 1 J=1,6
      DO 1 K=1,6
    1 U(J,K)=0.0
      U(1,1)=1.0
      U(2,2)=1.0
      U(3,3)=1.0
      U(4,4)=1.0
      U(5,5)=1.0
      U(6,6)=1.0
      U(4,3)=CTZ
      U(5,2)=-CLY
      U(6,1)=CLX
      RETURN
      END
C
C
      SUBROUTINE HANGEO (CTX,CLZ,CTY,U)
      DIMENSION U(6,6)
      DO 1 J=1,6
      DO 1 K=1,6
    1 U(J,K)=0.0
      U(1,1)=1.0
      U(2,1)=CTX
      U(2,2)=1.0
      U(3,3)=1.0
      U(4,4)=1.0
      U(5,4)=CTY
      U(5,5)=1.0
      U(6,3)=-CLZ
```

```
      U(6,6)=1.0
      RETURN
      END
C
C
      SUBROUTINE CFIELO (AJT,R,Z,G,SDRI,RHO,THETA,D,DI,FFAC,A)
      DIMENSION  A(6,6)
      PHI=THETA/Z
      RHOD=RHO
      IF(RHO-0.0)11,11,12
   11 V=-1.0
      GO TO 1
   12 V=1.0
    1 CP=COSF(PHI)
      SP=SINF(PHI)
      W=1.0/( G *AJT)
      F1=(PHI*CP+SP)/2.0
      F3=(SP-CP*PHI)/2.0
      F5=(2.0-2.0*CP-PHI*SP)/2.0
      F6=(2.0*PHI+PHI*CP-3.0*SP)/2.0
      RHO=ABSF(RHO)
      DO 2 J=1,6
      DO 2 K=1,6
    2 A(J,K)=0.0
      IF(SDRI)13,13,14
   13 AREA=(3.141592654/4.0)*(D**2-DI**2)*FFAC
      SD=RHO*THETA/(G*Z*AREA)
      GO TO 15
   14 SD=0.0
   15 A(1,1)=CP
      A(1,2)=(W*RHO*F1)-(RHO*F3*R)
      A(1,4)=V*SP
      A(1,5)=V*(W+R)*(RHO*PHI*SP)/2.0
      A(1,6)=V*(W+R)*(RHO**2)*F3
      A(2,2)=CP
      A(2,5)=V*SP
      A(2,6)=V*RHO*(1.0-CP)
      A(3,1)=-A(2,6)
      A(3,2)=-1.0*A(1,6)
      A(3,3)=1.0
      A(3,4)=RHO*SP
      A(3,5)=((R*(RHO**2)*PHI*SP)/2.0)-W*RHO**2*F5
      A(3,6)=(R*RHO**3*F3)-(W*F6*RHO**3)-SD
      A(4,1)=-V*SP
      A(4,2)=-V*(W+R)*RHO*PHI*SP/2.0
      A(4,4)=CP
      A(4,5)=(R*RHO*F1)-(W*RHO*F3)
      A(4,6)=A(3,5)
      A(5,2)=-V*SP
      A(5,5)=CP
      A(5,6)=A(3,4)
      A(6,6)=1.0
      RHO=RHOD
      RETURN
      END
C
```

```
C
      SUBROUTINE POINO (AMU,AIX,AIY,OMEGA,TL,Z,SDRI,B)
      DIMENSION B(6,6)
      AM=AMU*TL/(Z-1.0)
      DO 1 J=1,6
      DO 1 K=1,6
    1 B(J,K)=0.0
      B(1,1)=1.0
      B(2,1)=-AM*(AIX*OMEGA)**2
      B(2,2)=1.0
      B(3,3)=1.0
      B(4,4)=1.0
      IF(SDRI-0.)9,9,10
    9 B(5,4)=-AM*(AIY*OMEGA)**2
      GO TO 11
   10 B(5,4)=0
   11 B(5,5)=1.0
      B(6,3)=AM*OMEGA**2
      B(6,6)=1.0
      RETURN
      END
C
C
      SUBROUTINE CFIELD (R,Z,G,SD,RHO,THETA,D,DI,E,A)
      DIMENSION A(6,6)
      PHI=THETA/Z
      RD=ABSF(RHO)
      IF(RHO)1,1,2
    1 V=-1.0
      GO TO 3
    2 V=1.0
    3 CP=COSF(PHI)
      SP=SINF(PHI)
      DO 4 J=1,6
      DO 4 K=1,6
    4 A(J,K)=0.0
      A(1,1)=CP
      A(1,2)=-V*SP
      A(1,3)=-V*RD*(1.0-CP)
      A(1,4)=-V*RD**2*R*(PHI-SP)
      AR=3.141592654*(D**2-DI**2)/4.0
      W=1.0/(E*AR)
      F3=(SP-PHI*CP)*.5
      F1=(PHI*CP+SP)*.5
      F5=(2.0-2.0*CP-PHI*SP)*.5
      F6=(2.0*PHI+PHI*CP-3.0*SP)*.5
      A(1,5)=V*(RD*W*PHI*SP*.5-R*F5*RD**3)
      A(1,6)=RD*W*F1+R*F6*RD**3
      A(2,1)=V*SP
      A(2,2)=CP
      A(2,3)=RD*SP
      A(2,4)=R*RD**2*(1.0-CP)
      A(2,5)=RD*F3*(W+R*RD**2)
      A(2,6)=A(1,5)
      A(3,3)=1.0
      A(3,4)=RD*R*PHI
```

```
      A(3,5)=A(2,4)
      A(3,6)=A(1,4)
      A(4,4)=1.0
      A(4,5)=A(2,3)
      A(4,6)=A(1,3)
      A(5,5)=CP
      A(5,6)=A(1,2)
      A(6,5)=A(2,1)
      A(6,6)=CP
      RETURN
      END
C
C
      SUBROUTINE POINT (AMU,AIY,OMEGA,TL,Z,RI,B)
      DIMENSION B(6,6)
      AM=AMU*TL/(Z-1.)
      DO 1 J=1,6
      DO 1 K=1,6
    1 B(J,K)=0.0
      B(1,1)=1.0
      B(2,2)=1.0
      B(3,3)=1.0
      IF(RI)2,2,3
    2 B(4,3)=-AM*(AIY*OMEGA)**2
    3 B(4,4)=1.0
      B(5,2)=AM*OMEGA**2
      B(5,5)=1.0
      B(6,1)=-B(5,2)
      B(6,6)=1.0
      RETURN
      END
C
C
C
C
C
      SUBROUTINE SFIELD (DI,TL,R,Z,G,SD,D,E,FFAC,A)
      DIMENSION A(6,6)
      ARA=(3.141592654/4.0)*(D**2-DI**2)
      AREA=ARA*FFAC
      DL =TL/Z
      DO 1 J=2,5
    1 A(1,J)=0.0
      A(1,1)=1.0
      A(1,6)=DL/(E*ARA)
      A(2,1)=0.0
      A(2,2)=1.0
      A(2,3)=DL
      A(2,4)=DL**2*R/2.0
      IF(SD)2,2,3
    2 A(2,5)=DL**3*R/6.0-DL/(G*AREA)
      GO TO 4
    3 A(2,5)=DL**3*R/6.0
    4 A(2,6)=0.0
      A(3,1)=0.0
      A(3,2)=0.0
```

```
      A(3,3)=1.0
      A(3,4)=DL*R
      A(3,5)=A(2,4)
      A(3,6)=0.0
      DO 5 J=1,6
      A(4,J)=0.0
      A(5,J)=0.0
    5 A(6,J)=0.0
      A(4,4)=1.0
      A(4,5)=DL
      A(5,5)=1.0
      A(6,6)=1.0
      RETURN
      END
C
C
      SUBROUTINE SFIELO (DI,TL,AJT,R,Z,G,SDRI,D,FFAC,A)
      DIMENSION A(6,6)
      AREA=(3.141592654/4.0)*(D**2-DI**2)*FFAC
      DL=TL/Z
      DO 2 J=1,6
      DO 2 K=1,6
    2 A(J,K)=0.0
      A(1,1)=1.0
      A(2,2)=1.0
      A(3,3)=1.0
      A(4,4)=1.0
      A(5,5)=1.0
      A(6,6)=1.0
      IF(SDRI-0.)7,7,8
    7 A(3,6)=DL**3*R/6.0-DL/(G*AREA)
      GO TO 1
    8 A(3,6)=DL**3*R/6.0
    1 A(1,2)=DL/(G*AJT)
      A(3,4)=DL
      A(3,5)=DL**2*R/2.0
      A(4,5)=DL*R
      A(4,6)=A(3,5)
      A(5,6)=DL
      RETURN
      END
C
C
      SUBROUTINE MATMUL (A,B,U)
      DIMENSION A(6,6),B(6,6),C(6,6),U(6,6)
      DO 1 J=1,6
      DO 1 K=1,6
      C(J,K)=0.0
      DO 1 L=1,6
    1 C(J,K)=C(J,K)+B(J,L)*A(L,K)
      DO 2 J=1,6
      DO 2 K=1,6
    2 U(J,K)=C(J,K)
      RETURN
      END
C
```

```
C
      SUBROUTINE FINBRA(U,VV,V,MAT,A,Z)
      DIMENSION U(6,6),A(6,6),VV(6,3),V(6,3)
      TYPE DOUBLE VV,V
      IF(MAT) 65,64,65
   64 DO 67 J=1,6
      DO 67 K=1,3
      V(J,K)=0.
      DO 67 L=1,6
   67 V(J,K)=V(J,K)+U(J,L)*VV(L,K)
      RETURN
   65 N=Z-1.
      DO 5 M=1,N
      DO 4 J=1,6
      DO 4 K=1,3
      V(J,K)=0.
      DO 4 L=1,6
    4 V(J,K)=V(J,K)+U(J,L)*VV(L,K)
      DO 5 J=1,6
      DO 5 K=1,3
    5 VV(J,K)=V(J,K)
      DO 7 J=1,6
      DO 7 K=1,3
      V(J,K)=0.
      DO 7 L=1,6
    7 V(J,K)=V(J,K)+A(J,L)*VV(L,K)
      RETURN
      END
C
C
      SUBROUTINE FINMAT (U,UU,V,MAT,A,Z)
      DIMENSION U(6,6), A(6,6),V(6,3),UU(6,3)
      TYPE DOUBLE UU,V
      IF(MAT) 65,64,65
   64 DO 67 J=1,6
      DO 67 K=1,3
      V(J,K)=0.
      DO 67 L=1,6
   67 V(J,K)=V(J,K)+U(J,L)*UU(L,K)
      RETURN
   65 N=Z-1.
      DO 5 M=1,N
      DO 4 J=1,6
      DO 4 K=1,3
      V(J,K)=0.
      DO 4 L=1,6
    4 V(J,K)=V(J,K)+U(J,L)*UU(L,K)
      DO 5 J=1,6
      DO 5 K=1,3
    5 UU(J,K)=V(J,K)
      DO 7 J=1,6
      DO 7 K=1,3
      V(J,K)=0.
      DO 7 L=1,6
    7 V(J,K)=V(J,K)+A(J,L)*UU(L,K)
      RETURN
```

```
      END
C
C
      SUBROUTINE BRANCH (R3,VV,PHI,U)
      DIMENSION R3(3,3),VV(6,3),U(6,6),R1(25,25),LL(25),MM(25),R2(3,3),
     1R(3,3),G1(3,3),G2(3,3)
      TYPE DOUBLE VV
      DO 4 L=1,3
      R1(1,L)=VV(1,L)
      R1(2,L)=VV(2,L)
      R1(3,L)=VV(3,L)
      R2(1,L)=VV(4,L)
      R2(2,L)=VV(5,L)
    4 R2(3,L)=VV(6,L)
      SP=SINF(PHI)
      CP=COSF(PHI)
      G1(1,1)=CP
      G1(1,2)=-SP
      G1(1,3)=0.
      G1(2,1)=SP
      G1(2,2)=CP
      G1(2,3)=0.
      G1(3,1)=0.
      G1(3,2)=0.
      G1(3,3)=1.
      G2(1,1)=1.
      G2(1,2)=0.
      G2(1,3)=0.
      G2(2,1)=0.
      G2(2,2)=CP
      G2(2,3)=-SP
      G2(3,1)=0.
      G2(3,2)=SP
      G2(3,3)=CP
      CALL INVERT (R1,3,D,LL,MM)
      DO 5 J=1,3
      DO 5 K=1,3
      R3(J,K)=0.0
      DO 5 L=1,3
    5 R3(J,K)=R3(J,K)+R1(J,L)*G1(L,K)
      DO 6 J=1,3
      DO 6 K=1,3
      R1(J,K)=0.0
      DO 6 L=1,3
    6 R1(J,K)=R1(J,K)+G2(J,L)*R2(L,K)
      DO 7 J=1,3
      DO 7 K=1,3
      R(J,K)=0.0
      DO 7 L=1,3
    7 R(J,K)=R(J,K)+R1(J,L)*R3(L,K)
      DO 8 J=1,6
      DO 8 K=1,6
    8 U(J,K)=0.0
      U(1,1)=1.0
      U(2,2)=1.0
      U(3,3)=1.0
```

```
      U(4,4)=1.0
      U(5,5)=1.0
      U(6,6)=1.0
      DO 9 L=1,3
      U(4,L)=R(1,L)
      U(5,L)=R(2,L)
    9 U(6,L)=R(3,L)
      RETURN
      END
C
C
      SUBROUTINE BRANCO (R3,VV,PHI,U)
      DIMENSION R3(3,3),U(6,6),VV(6,3),R1(25,25),R2(3,3),
      TYPE DOUBLE VV
     1G1(3,3),G2(3,3),S(3,3),LL(25),MM(25)
      DO 4 K=1,3
      R1(1,K)=VV(1,K)
      R1(2,K)=VV(3,K)
      R1(3,K)=VV(4,K)
      R2(1,K)=VV(2,K)
      R2(2,K)=VV(5,K)
    4 R2(3,K)=VV(6,K)
      SP=SINF(PHI)
      CP=COSF(PHI)
      G1(1,1)=CP
      G1(1,2)=0.
      G1(1,3)=-SP
      G1(2,1)=0.
      G1(2,2)=1.
      G1(2,3)=0.
      G1(3,1)=SP
      G1(3,2)=0.
      G1(3,3)=CP
      G2(1,1)=CP
      G2(1,2)=SP
      G2(1,3)=0.
      G2(2,1)=-SP
      G2(2,2)=CP
      G2(2,3)=0.
      G2(3,1)=0.
      G2(3,2)=0.
      G2(3,3)=1.
      CALL INVERT (R1,3,D,LL,MM)
      DO 5 J=1,3
      DO 5 K=1,3
      R3(J,K)=0.0
      DO 5 L=1,3
    5 R3(J,K)=R3(J,K)+R1(J,L)*G1(L,K)
      DO 6 J=1,3
      DO 6 K=1,3
      R1(J,K)=0.0
      DO 6 L=1,3
    6 R1(J,K)=R1(J,K)+G2(J,L)*R2(L,K)
      DO 7 J=1,3
      DO 7 K=1,3
      S(J,K)=0.0
```

```
      DO 7 L=1,3
    7 S(J,K)=S(J,K)+R1(J,L)*R3(L,K)
      DO 8 J=1,6
      DO 8 K=1,6
    8 U(J,K)=0.0
      U(1,1)=1.0
      U(2,1)=S(1,1)
      U(2,2)=1.0
      U(2,3)=S(1,2)
      U(2,4)=S(1,3)
      U(3,3)=1.0
      U(4,4)=1.0
      U(5,1)=S(2,1)
      U(5,3)=S(2,2)
      U(5,4)=S(2,3)
      U(5,5)=1.0
      U(6,1)=S(3,1)
      U(6,3)=S(3,2)
      U(6,4)=S(3,3)
      U(6,6)=1.0
      RETURN
      END
C
C
      SUBROUTINE STATEM(SVI,UU,D1,D2,P1,P2,PP,PM,SS)
      DIMENSION SVI(6,1),UU(6,3)
      TYPE DOUBLE UU,P1,P2,D1,D2
      M=0
      DO 1 J=1,6
      DO 1 K=1,3
    1 UU(J,K)=0.
      IF(PM) 40,40,41
   40 DO 42 J=1,6
      IF(SVI(J,1)) 42,42,43
   43 M=M+1
      UU(J,M)=1.
      IF(M-1) 42,44,42
   44 UU(J,2)=-D1
      UU(J,3)=-D2
   42 CONTINUE
      RETURN
   41 P1=P1+D1
      P2=P2+D2
      IF(SS) 14,15,15
   15 IF(SS-1.) 12,11,10
   14 IF(PP-1.) 12,11,10
   10 DO 4 J=1,6
      IF(SVI(J,1)) 4,4,2
    2 M=M+1
      UU(J,M)=1.
      IF(M-2) 4,3,5
    3 UU(J,1)=P1
      GO TO 4
    5 UU(J,1)=P2
    4 CONTINUE
```

```
      RETURN
   11 DO 21 J=1,6
      IF(SVI(J,1)) 21,21,22
   22 M=M+1
      UU(J,M)=1.
      IF(M-1) 25,26,25
   26 UU(J,3)=P1
   25 IF(M-2) 21,23,21
   23 UU(J,3)=P2
   21 CONTINUE
      RETURN
   12 DO 31 J=1,6
      IF(SVI(J,1)) 31,31,32
   32 M=M+1
      UU(J,M)=1.
      IF(M-1) 35,36,35
   36 UU(J,2)=P2
   35 IF(M-2) 31,31,33
   33 UU(J,2)=P1
   31 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE STATEB(SVB,VV,N,YY,L,BRC,D1,D2,PP,PM,SS,Q1,Q2,Q3,FINK)
      DIMENSION  VV(6,3),YY(3,3,25),V(3,3),SVB(6,22)
      TYPE DOUBLE VV
      B1=0.
      B2=0.
      B3=0.
      D1=0.
      D2=0.
      IF(PM) 33,40,22
   22 IF(BRC) 21,20,21
   21 IF(L-1) 23,20,23
   23 DO 10 J=1,3
      DO 10 K=1,3
   10 V(J,K)=YY(J,K,N)
      IF(SS) 11,12,12
   11 IF(PP-1.) 32,31,30
   12 IF(SS-1.) 33,31,30
   30 B1=V(1,1)+V(1,2)*D1+V(1,3)*D2+B1
      B2=V(2,1)+V(2,2)*D1+V(2,3)*D2+B2
      B3=V(3,1)+V(3,2)*D1+V(3,3)*D2+B3
      GO TO 33
   31 B1=B1+V(1,1)*D2+V(1,3)*D1+V(1,2)
      B2=B2+V(2,1)*D2+V(2,3)*D1+V(2,2)
      B3=B3+V(3,1)*D2+V(3,3)*D1+V(3,2)
      GO TO 33
   32 B1=B1+V(1,1)*D1+V(1,2)*D2+V(1,3)
      B2=B2+V(2,1)*D1+V(2,2)*D2+V(2,3)
      B3=B3+V(3,1)*D1+V(3,2)*D2+V(3,3)
   33 Q1=Q1*B1
      Q2=B1*Q2+B2
      Q3=B1*Q3+B3
```

```
20 M=1
   DO 1 J=1,6
   DO 1 K=1,3
 1 VV(J,K)=0.0
   DO 4 J=1,6
   IF(SVB(J,N)) 4,4,2
 2 IF(M-1)7,3,7
 3 IF(PM) 34,35,35
35 VV(J,1)=Q1
   GO TO 36
34 VV(J,1)=1.
36 M=M+1
   GO TO 4
 7 IF(M-2) 4,8,9
 8 VV(J,1)=Q2
   VV(J,M)=1.
   M=M+1
   GO TO 4
 9 VV(J,1)=Q3
   VV(J,M)=1.
 4 CONTINUE
   RETURN
40 IF(FINK) 54,51,54
54 IF(BRC) 51,51,52
52 DO 53 J=1,3
   DO 53 K=1,3
53 V(J,K)=YY(J,K,N)
   D1=(V(1,2)/V(1,1)+V(2,2)/V(2,1)+V(3,2)/V(3,1))/3.
   D2=(V(1,3)/V(1,1)+V(2,3)/V(2,1)+V(3,3)/V(3,1))/3.
51 M=0
   DO 41 J=1,6
   DO 41 K=1,3
41 VV(J,K)=0.
   DO 42 J=1,6
   IF(SVB(J,N)) 42,42,43
43 M=M+1
   VV(J,M)=1.
   IF(M-1) 42,44,42
44 VV(J,2)=-D1
   VV(J,3)=-D2
42 CONTINUE
   RETURN
   END
C
C
   SUBROUTINE BRCOR(R3,UU,XX,JK,VV,PM)
   DIMENSION R3(3,3),UU(6,3),XX(3,3),UUU(3,3),VV(6,3)
   TYPE DOUBLE VV,UU
   IF(PM) 20,21,20
20 IF(JK) 10,10,11
10 DO 12 J=1,3
   DO 12 K=1,3
12 UUU(J,K)=UU(J,K)
   GO TO 3
11 DO 13 K=1,3
   UUU(1,K)=UU(1,K)
```

```
      UUU(2,K)=UU(3,K)
   13 UUU(3,K)=UU(4,K)
    3 DO 1 J=1,3
      DO 1 K=1,3
      XX(J,K)=0.
      DO 1 L=1,3
    1 XX(J,K)=XX(J,K)+R3(J,L)*UUU(L,K)
      RETURN
   21 IF(JK) 25,25,22
   25 DO 23 J=1,3
      DO 23 K=1,3
   23 XX(J,K)=VV(J,K)
      RETURN
   22 DO 24 K=1,3
      XX(1,K)=VV(1,K)
      XX(2,K)=VV(3,K)
   24 XX(3,K)=VV(4,K)
      RETURN
      END
C
C
      SUBROUTINE DELMA(UU,SVE,V,X2,Y2,KKK,X1,FF,PP,HH,RR,SS,Y1,REM,FINK,
     1PM,D1,D2)
      DIMENSION UU(6,3),V(6,3),SVE(6,1),D(3,3)
      TYPE DOUBLE UU,V,D,X1,X2,Y1,Y2,DV,D1,D2
      L=0
      X1=0.
      X2=0.
      Y2=0.
      Y1=0.
      D1=0.
      D2=0.
      DO 111 J=1,3
      DO 111 K=1,3
  111 D(J,K)=0.
      DO 91 N=1,6
      IF(SVE(N,1)) 91,2,91
    2 L=L+1
      DO 92 K=1,3
   92 V(L,K)=UU(N,K)
   91 CONTINUE
      IF(PM) 66,64,61
   64 IF(FINK) 63,62,63
   62 D1=(V(1,2)/V(1,1)+V(2,2)/V(2,1)+V(3,2)/V(3,1))/3.
      D2=(V(1,3)/V(1,1)+V(2,3)/V(2,1)+V(3,3)/V(3,1))/3.
      FINK=1.
      RETURN
   63 FINK=0.
      RETURN
   66 RETURN
   61 IF(SS) 311,312,312
  311 IF(PP-1.)400,401,402
  312 IF(SS-1.) 313,314,315
  402 IF(RR-1.) 300,403,300
  315 IF(RR-1.) 340,341,340
  300 IF(V(3,3)) 93,94,93
```

```
340 IF(V(3,2)) 93,94,93
 93 DO 95 J=2,3
    DO 95 K=1,3
 95 D(J,K)=V(J,K)
    DV=V(1,2)*D(2,3)-V(1,3)*D(2,2)
    IF(DV) 98,94,98
 94 RR=1.0
    HH=1.
    RETURN
403 IF(V(2,3)) 96,200,96
341 IF(V(2,2))96,200,96
 96 DO 97 K=1,3
    D(2,K)=V(3,K)
 97 D(3,K)=V(2,K)
    DV=V(1,2)*D(2,3)-V(1,3)*D(2,2)
    IF(DV) 98,200,98
 98 X1=X1-(V(1,1)*D(2,3)-V(1,3)*D(2,1))/DV
    X2=X2-(V(1,2)*D(2,1)-V(1,1)*D(2,2))/DV
    IF(X1) 100,102,100
102 IF(X2) 100,101,100
101 FF=1.0
    RETURN
100 IF(SS) 354,355,355
354 Y2=Y2-D(3,1)/D(3,3)-X1*D(3,2)/D(3,3)
    RETURN
355 Y1=Y1-D(3,1)/D(3,2)-X2*D(3,3)/D(3,2)
    RETURN
 99 IF(SS) 20,21,21
 20 PP=1.
    GO TO 23
 21 SS=1.
 23 RR=0.
    HH=1.
    RETURN
314 IF(RR-1.) 330,331,330
401 IF(RR-1.0) 301,404,301
330 IF(V(3,1)) 81,82,81
301 IF(V(3,2)) 81,82,81
 81 DO 83 J=2,3
    DO 83 K=1,3
 83 D(J,K)=V(J,K)
    DV=V(1,1)*D(2,2)-V(1,2)*D(2,1)
    IF(DV) 86,82,86
 82 RR=1.0
    HH=1.
     RETURN
331 IF(V(2,1)) 84,70,84
404 IF(V(2,2)) 84,70,84
 84 DO 85 K=1,3
    D(2,K)=V(3,K)
 85 D(3,K)=V(2,K)
    DV=V(1,1)*D(2,2)-V(1,2)*D(2,1)
    IF(DV) 86,70,86
 86 X1=X1-(V(1,3)*D(2,2)-V(1,2)*D(2,3))/DV
    X2=X2-(V(1,1)*D(2,3)-V(1,3)*D(2,1))/DV
    IF(X1)203,201,203
```

```
201 IF(X2) 203,202,203
202 FF=1.0
    RETURN
203 IF(SS)352,353,353
353 Y1=Y1-D(3,3)/D(3,1)-X2*D(3,2)/D(3,1)
    RETURN
352 Y2=Y2-D(3,3)/D(3,2)-X1*D(3,1)/D(3,2)
    RETURN
 70 IF(SS) 25,24,24
 24 SS=2.
    GO TO 26
 25 PP=2.
 26 RR=0.
    HH=1.
    RETURN
313 IF(RR-1.) 320,321,320
400 IF(RR-1.) 302,405,302
320 IF(V(3,3)) 71,72,71
302 IF(V(3,1))71,72,71
 71 DO 73 J=2,3
    DO 73 K=1,3
 73 D(J,K)=V(J,K)
    DV=V(1,1)*D(2,3)-V(1,3)*D(2,1)
    IF(DV) 76,72,76
 72 RR=1.0
    HH=1.
    RETURN
405 IF(V(2,1)) 74,99,74
321 IF(V(2,3)) 74,99,74
200 KKK=1
    RR=0.
    RETURN
 74 DO 75 K=1,3
    D(2,K)=V(3,K)
 75 D(3,K)=V(2,K)
    DV=V(1,1)*D(2,3)-V(1,3)*D(2,1)
    IF(DV) 76,99,76
 76 X2=X2-(V(1,2)*D(2,3)-V(1,3)*D(2,2))/DV
    X1=X1-(V(1,1)*D(2,2)-D(2,1)*V(1,2))/DV
    IF(X1) 213,211,213
211 IF(X2) 213,212,213
212 FF=1.0
    RETURN
213 IF(SS) 350,351,351
351 Y1=Y1-(D(3,2)+X2*D(3,1))/D(3,3)
    RETURN
350 Y2=Y2-(D(3,2)+X1*D(3,3))/D(3,1)
    RETURN
    END
```

APPENDIX E

ACCURACY OF METHODS & SAMPLE PROBLEMS

E-1 General remarks.

Appendix E contains the results of vibration analyses of several piping systems. The first group was performed to establish the accuracy of the methods and compare them with the VIPIPE output. The second group was performed to establish assurance of solution by comparing the natural frequencies got from various methods.

E-2 Method accuracy analyses.

E-2-1 Straight sections.

End conditions.

System 1 - fixed-fixed.

System 2 - fixed-free.

System 3 - fixed-propped.

System parameters (for system 1,2,3).

length - 200 inches.

diameter - 2.0 inches.

wall thickness - 0.125 inches.

Intensive properties (for system 1,2,3)

density - 490 lbs/cu-ft

shear modulus - 12×10^6 psi

elastic modulus - 30×10^6 psi

Mathematical model: Distributed mass with the effect of shear deflection and rotational inertia neglected to permit comparison with classical solution.

Comparison of results.

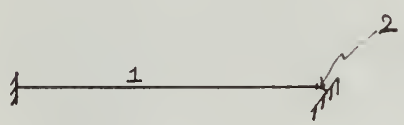
As explained in Appendix B-9-4, when the iteration reaches the solution acceptability criterion, the corresponding natural frequency is determined using linear interpolation. Theoretically exact comparison values are obtained by use of the same iteration method with the solution acceptability criterion as 0.0007 rad/sec. The values obtained using VIPIPE have a solution acceptability criterion 0.0185 rad/sec. Comparison is made by percent difference.

The M-D method fails the systems 1 through 3, because the purging factors turn out to be all zeros. So for the purpose of testing the M-D method, modified System 1 is provided below.

The letters D or S in parentheses denote double precision and single precision respectively.

Modified System 1:

Add a very small curved section to the System 1. Second section is a mathematical model which has the same property as System 1 except the system parameters.



Second section parameters:

included angle of arc - 45 degrees

radius of curvature - 0.01 inches

(This value of radius of curvature is not possible in an actual system, since the radius of curvature less than the radius of the pipe. In effect subroutine STIFCO or STIFOO will provide the transfer matrix of a massless rigid corner).

Results and comparison.*

a. System 1 and modified System 1.** (in-plane).

Mode No.	Freq.(rad/sec) and difference in %.			
	Comparison values	Delta m. (D)	P-M.-A(D)	P-Mm.-F(D)
1	75.09994540	75.09994544 0 %	75.09994550 0 %	75.09994571 0 %
2	207.01589137	207.01589036 5.0×10^{-7}%	207.01589094 0 %	207.01589228 4.4×10^{-7} %
3	405.83391916	405.83393183 3.0×10^{-6}%	405.83403022 2.4×10^{-5}%	405.83394702 7.0×10^{-7}%
4	670.86408383	670.86310856 2.8×10^{-4}%	670.86341454 1.0×10^{-5}%	670.86345124 9.1×10^{-5}%
5	1002.15520225	1002.19978814 4.7×10^{-3}%	1002.18735784 3.2×10^{-3}%	1002.19300449 3.8×10^{-3}%
6	1399.70436269		1399.59981662 7.5×10^{-3}%	1399.64448571 4.3×10^{-3}%
7	1863.51172599			1959.59556134 5.1 %

Mode No.	Freq.(rad.sec) and difference in %			
	Delta m. (S)	P-M m.-A(S)	M-D m.(D)***	M-D m.(S)***
1.	75.09994535 0 %	75.09994865 4.0×10^{-6} %	75.09994539 0 %	75.09994534 0 %
2	207.01589228 5.0×10^{-8}%	207.01589071 0 %	207.01588836 1.5×10^{-6} %	207.01588891 0 %
3	405.83398145 1.5×10^{-6}%	405.83403455 2.1×10^{-5} %	405.83395151 9.5×10^{-6} %	405.83401120 2.0×10^{-5} %
4	670.86345124 7.5×10^{-5}%	670.86256288 2.2×10^{-4} %	670.86374475 5.1×10^{-5} %	670.86118054 4.3×10^{-4} %
5		1002.23499946 8.0×10^{-3} %	1002.19909306 4.4×10^{-4} %	1002.21779856 6.3×10^{-3} %
6		1399.17084415 4.5×10^{-2} %	1399.94044241 3.4×10^{-2} %	

*Sources of comparison frequencies for system 1 through 3 are given in E-2-2.

(Additional footnotes on next page).

**In all the tables which follow, simple and evident abbreviations are used. For example, P-M m.-A(D) means the P-M method, sub-procedure A in double precision arithmetic.

***Modified System 1.

b. System 2.(in-plane)

Mode NO	Freq.(rad/sec) and difference in %			
	Comparison values	Delta m. (D)	P-M m.-A (D)	P-M m.-F (D)
1	11.80213586	11.80213389 5.5×10^{-5} %	11.80214614 9.5×10^{-5} %	11.80213746 1.4×10^{-5} %
2	73.96272297	73.96272167 1.7×10^{-6} %	73.96272478 1.7x10-6 %	73.96272367 0 %
3	207.09776595	207.09776685 4.1×10^{-8} %	207.09776388 1.1×10^{-6} %	207.09776562 0 %
4	405.82896584	405.82894132 6.1×10^{-6} %	405.82893748 7.0×10^{-6} %	405.82900191 9.1×10^{-6} %
5	670.86435913	670.86200817 2.2x10-5 %	670.86326697 1.7×10^{-5} %	670.86325735 1.7x10-5 %
6	1002.1551877	1002.26833752 1.1x10-2 %	1002.18809484 4.4×10^{-3} %	1002.19715557 4.2×10^{-3} %
7	1399.70436329	1399.08849534 2.0×10^{-2} %	1399.61231065 6.6×10^{-3} %	1399.14599562 4.2×10^{-3} %
8	1863.51172626		1862.55983728 5.1×10^{-2} %	

c. System 3.(in-plane)

Mode NO	Freq.(rad/sec) and difference in %			
	Comparison values	Delta m. (D)	P-M m.-A (D)	P-M m.-F (D)
1	51.75397285	51.75397263 0 %	51.75397569 0 %	51.75397564 0 %
2	167.71502090	167.71602009 0 %	167.71602200 0 %	167.71602189 0 %
3	349.92609071	349.92608231 1.1×10^{-7} %	349.92610259 3.4×10^{-6} %	349.92610209 3.4x10-6 %
4	598.39432089	598.39381063 8.5×10^{-5} %	598.39446341 2.5×10^{-5} %	598.39407344 4.2×10^{-5} %
5	913.12074582	913.13150640 1.2×10^{-3} %	913.13180993 1.2×10^{-3} %	913.13109335 1.2×10^{-3} %
6	1294.10536551	1293.77838653 2.7×10^{-2} %	1294.11223844 5.8×10^{-4} %	1294.11761177 1.9×10^{-4} %
7	1741.34817946		1748.59011823 4.2×10^{-1} %	

E-2-2 Sources of comparison frequencies for system 1 through 3.

The governing fourth order equation of a homogeneous single component straight system

$$y^{IV} = \frac{\mu \omega^4}{E \cdot I} \cdot y$$

was solved to give

$$y = a \cos \alpha x + b \cosh \alpha x + c \sin \alpha x + d \sinh \alpha x$$

Upon applying boundary conditions for System 1 ($y = y' = 0$ at $x = 0$, $x = L$) and simplifying, it was found that the eigenvalues of the system must satisfy

$$\cos \alpha \cdot \cosh \alpha = 1 \qquad \text{(E-2-2-1)}$$

where

$$\alpha = \sqrt[4]{\frac{\mu \omega^4}{E I}} \cdot L \qquad \text{(E-2-2-2)}$$

Upon applying the boundary conditions for System 2($y = y' = 0$ at $x = 0$, $y'' = y''' = 0$ at $x = L$), it was found that the eigenvalues of the system must satisfy

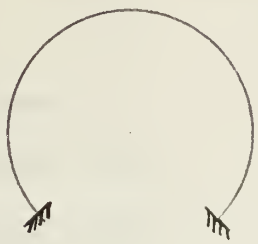
$$\cos \alpha \cdot \cosh \alpha = -1 \qquad \text{(E-2-2-3)}$$

Upon applying the boundary conditions of System 3($y = y' = 0$ at $x = 0$, $y = y'' = 0$ at $x = L$), it was found that the eigenvalues must satisfy

$$\sin \alpha \cdot \cosh \alpha - \cos \alpha \cdot \cosh \alpha = 0 \qquad \text{(E-2-2-4)}$$

The roots of equation E-2-2-1, E-2-2-3 and E-2-2-4, from which the natural frequencies of the corresponding systems may easily be found, were obtained through iteration using a digital computer.

E-2-3 Curved section (System 4)



System parameters:

radius of curvature	50 inches
included angle of arc	270 degrees
diameter	2.0 inches
wall thickness	.125 inches

Intensive properties:

density	556. lbs/cu-ft
shear modulus	12×10^6 psi
elastic modulus	30×10^6 psi

End conditions: fixed-fixed.

Mathematical model: lumped mass with shear deflection and rotational inertia neglected.

Results.

a. In-plane vibration*

Mode No.	Frequency (rad/sec)			
	Delta m.(D)	M-D m. (D)	P-M m.-A (D)	P-M m.-C (D)
1	66.69934902	66.69934902	66.69934980	66.69934948
2	171.22352538	171.22352538	171.22352610	171.22352592
3	335.35007083	335.35007083	335.35007083	335.35007174
4		535.64529414	535.6429665	535.64529637
5		773.44358487	773.44358842	773.44358812

b. Out-of-plane vibration*

Mode No.	Frequency (rad/sec)		
	Delta m.(D)	M-D m. (D)	P-M m.-A (D)
1	35.36309735	35.36309735	35.36309778
2	94.42588921	94.42588921	94.42588653
3	207.89287257	207.89278647	207.89287345
4	369.11199342	369.11278647	369.11278689
5		571.22235666	571.22235577

*See footnote next page.

*The fundamental frequency value of in-plane and out-of-plane case are close to the values obtained by employing equation $\omega = F(\alpha) \cdot (EI/\mu R^4)^{\frac{1}{2}}$ after obtaining $F(\alpha)$ from the curves in reference [5]. Natural frequency values from the three methods are the same up to six digits. One can get more higher mode frequencies using the M-D and P-M method.

E-3 Other Sample Problems

A. System 5

System parameters

Parameter	component						
	1	2	3	4	5	6	7
diameter (in)	2.0	2.0	2.0	H	2.0	2.0	2.0
				A			
wall thickness (in)	0.75	0.75	0.75	N	0.75	0.75	0.75
				G			
length (in)	130.	1.40	6.0	E	60.21	1.50	6.28
				R			
radius of curve (in)		4.0			30.		4.0
incl. angle of arc		20.			115.		90.

Intensive properties of all components except hangers:

density: 556 lbs/cu-ft.*

shear modulus: 12×10^6 psi

elastic modulus: 30×10^6 psi

Hanger spring rates:

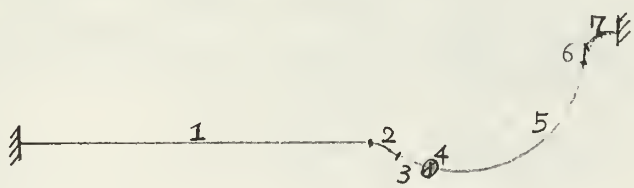
CLX , CLY , CLZ 1000lbs/in.

CTX , CTY , CTZ 1000 in-lb/rad.

End conditions:

both ends of system fixed.

Mathematical model: Distributed mass with the effects of shear deflection and rotational inertia considered.



*This fictitious value was chosen to permit comparing with results of Fink [4] who used the same value.

Results. (in-plane vibration)

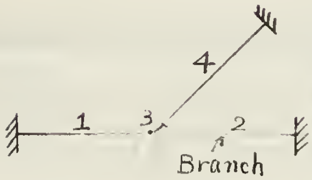
Mode NO	Frequency(rad/sec)			
	Delta m. (D)	M-D m. (D)	P-M m.-A (D)	P-M m.-C (D)
1	116.89479180			
2	304.88354398			
3	566.43670976			
4		826.88282292	826.88281950	826.88281929
5		1126.83773518	1126.83799669	1126.83797508
6		1666.58935151	1666.58812881	1666.58878192
7		2099.54173011	2099.5352509	2099.52844030

Results in single precision arithmetic and comparison with double precision results.

Mode NO	Frequency(rad/sec) and diff. in %			
	Delta m. (S)	M-D m. (S)	P-M m.-A (S)	P-M m.-C (S)
1	116.89479181 0 %			
2	304.88354658 1.0×10^{-6} %			
3	566.43620786 8.9×10^{-5} %			
4		826.88281494 1.0×10^{-6} %	826.88281417 6.0×10^{-7} %	826.88281764 1.9×10^{-7} %
5		1126.83768564 4.4×10^{-6} %	1126.83786333 1.2×10^{-5} %	1126.83745223 4.6×10^{-5} %
6		1666.58927426 4.7×10^{-6} %		
7		2099.53091794 5.0×10^{-3} %		

B. System 6 (single branch system)

System parameters:



parameter	components 1	2	3	4
dia. (inch)	2.0	2.0	2.0	2.0
wall thickness (inch)	.125	.125	.125	.125
length (inch)	100.	100.	.01	100.
incl. angle of arc. (degree)	0	0	45	0

Intensive properties:

density 556 lbs/cu-ft

shear modulus 12×10^6 psi

elastic modulus 30×10^6 psi

End conditions: All ends fixed.

Mathematical model: Distributed mass with the effects of shear deflection and rotational inertia neglected.

Results

Mode No.	Frequency (rad/sec) Delta m.(D)	M-D m. (D)	P-M m.-A (D)	P-M m.-C (D)
1	194.25470558			
2	278.61078241			
3	628.59556355			
4	747.68138190	747.68087208	747.68137310	747.68137759
5		1300.29478437	1300.31646085	1300.31641281
6		1381.79339427	1381.86234251	1381.86230734
7		1991.80356348	1991.38946110	1991.38946101
8		2257.92926800	2257.58161443	2257.58016324

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	22
2.	Library U. S. Naval Postgraduate School Monterey, California	2
3.	Mechanical Engineering Department U. S. Naval Postgraduate School Monterey, California	1
4.	Prof. J. E. Brock Department of Mechanical Engineering U. S. Naval Postgraduate School Monterey, California	2
5.	Library Republic of Korea Naval Academy Chin Hae, Korea	2
6.	LT Yong Su Kim, ROKN #714-7 RoSong Dong, Chun Ju Republic of Korea	2
7.	LT C. R. Dunlap III SMC #1025 U. S. Naval Postgraduate School Monterey, California	1
8.	Mr. John Y. Rhee #1121 McClelln Ave., Monterey, California	1

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*	2a. REPORT SECURITY CLASSIFICATION
U. S. Naval Postgraduate School Monterey, California	Unclassified 2b. GROUP N.A.

3. REPORT TITLE

Vibration Analysis of Piping Systems

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

None

5. AUTHOR(S) *(Last name, first name, initial)*

KIM, Yong Su, LT, ROKN

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
May 1966	130	7

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
N.A.	N.A.
b. PROJECT NO. N.A.	
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)* N.A.
d.	

10. AVAILABILITY/LIMITATION NOTICES

#1 [illegible] 12/22/69

~~Qualified requesters may obtain copies of this report from DDC.~~

This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
N.A.	U. S. Navy

13. ABSTRACT

A digital computer program, originally developed by George E. Fink, capable of determining the in-and/or out-of-plane vibration frequencies of a single plane piping system, using the basic transfer method, is modified and augmented by the writer. The program is modified to use the writer's method to reduce the amount of calculation and is augmented to use Marguerre and Uhrig's modifying frequency determinant method and Pestel and Mahrenholtz's remainder method for higher natural frequencies.

An analysis is made of difficulties encountered with the original transfer method and utilizing the modified methods.

The program accepts branched systems and non-still intermediate supports. A discussion and an explanation of how to use the program are included.

DD FORM 1 JAN 64 1473

Unclassified

Security Classification

Unclassified

Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Piping Systems						
Vibration Analysis						
Computer Program						
Transfer Matrix						
Natural Frequencies						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through __________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through __________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through __________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.

DD FORM 1 JAN 64 1473 (BACK)

Unclassifed

Security Classification